

SISTEM PAKAR MENDETEKSI PENYAKIT ASMA PADA PUSKESMAS LINGKAR TIMUR BENGKULU

Asnawati¹, Rita Prima Bendriyanti², Hari Aspriyono³

Dosen Tetap Fakultas Ilmu Komputer Universitas Dehasen Bengkulu

ABSTRACT

Expert system detection of asthma. This research aims to create an expert system application detection asthma in eastern rim Bengkulu clinic using Visual Basic 6.0 programming language in knowing gejala and without medication should consult with relevant dokter or midwife to make the diagnosis of a patient.

Data collection methods used in the making of this observation, library research and interviews, the results of research conducted, it can be concluded that the detection expert system asthma in eastern rim centers using Visual Basic 6.0 programming language can improve the performance of the health center east rim Bengkulu

INTISARI

Sistem pakar deteksi penyakit asma pada puskesmas lingkaran timur. Penelitian ini bertujuan untuk membuat aplikasi sistem pakar deteksi penyakit asma pada puskesmas lingkaran timur Bengkulu dengan menggunakan Bahasa Pemrograman Visual Basic 6.0 dalam mengetahui gejala dan obat tanpa harus konsultasi dengan dokter atau bidan terkait untuk melakukan diagnosa dari seorang pasien.

Metode pengumpulan data yang digunakan dalam penelitian ini yaitu observasi, studi pustaka dan wawancara, dari hasil penelitian yang dilakukan maka dapat disimpulkan bahwa sistem pakar deteksi penyakit asma pada puskesmas lingkaran timur dengan menggunakan Bahasa pemrograman Visual basic 6.0 dapat meningkatkan kinerja dari puskesmas lingkaran timur Bengkulu.

Kata kunci : Sistem Pakar, Visual Basic

I. PENDAHULUAN

Kemajuan teknologi komputer dewasa ini mampu membangkitkan pengaruh yang besar terhadap kemajuan bidang-bidang lain seperti pertanian, fabrikasi, kedokteran dan lain sebagainya. Betapa tidak, komputer yang pada awalnya merupakan benda mati dan hanya berupa alat bantu perhitungan dan pengetikan, kini telah mampu bekerja sebagai alat untuk melakukan analisis terhadap sejumlah data yang rumit menjadi informasi yang jelas dan mudah dipahami. Yang mana semua itu karena didukung oleh kemajuan teknologi informasi dengan komputer sebagai alat bantu dalam pengolahan datanya.

Teknologi Informasi yang semakin hari semakin dibutuhkan oleh banyak pengguna, tentunya menuntut terus menerus untuk dapat memberikan berbagai gambaran dan uraian yang

jelas terhadap informasi yang diberikan. Sehingga semua informasi yang diterima sesuai dengan yang diharapkan oleh pengguna.

Kecerdasan buatan yang merupakan salah satu bidang ilmu yang menekuni teknologi informasi banyak menyumbang berbagai teknik pengolahan data yang diselesaikan dengan teknik-teknik yang ada di dalam bidang ilmu kecerdasan buatan. Dimana bidang-bidang ilmu yang masih menjadi cakupan dari kecerdasan buatan adalah Data Mining, Sistem Pakar, Logika Fuzzy, Jaringan Saraf Tiruan dan lain sebagainya.

Sistem Pakar (*Expert systems*) merupakan ilmu yang mempelajari berbagai rekaman data dan fakta-fakta yang ada yang diberikan/diungkapkan oleh seorang pakar. Sehingga program selanjutnya dapat digunakan untuk menggantikan keberadaan

pakar. Terlebih lagi karena keberadaan seorang dokter yang dapat saja tidak berada di tempat pada saat dibutuhkan analisisnya. Maka program yang dapat bertindak sebagai seorang pakar menjadi sangat diperlukan.

Penyakit dengan berbagai aspek dan gejala yang dirasakan dan akibat yang ditimbulkannya perlu mendapat gambaran yang jelas terhadap indikasi yang ada, sehingga paramedis dan/atau dokter nantinya tidak akan salah dalam melakukan diagnosa terhadap berbagai gejala yang diberikan oleh pasien untuk ditetapkan jenis penyakit yang dideritanya.

Sehingga pada diagnosa yang salah atau kurang tepat akan menyebabkan kesalahan dalam penanganan yang akan dilakukan oleh paramedis seperti nama penyakit, cara penanganan dini dan obat-obatan yang harus dikonsumsi. Karena tidak jarang seorang dokter dan paramedis lain, salah dalam melakukan diagnosa yang mengakibatkan dampak yang cukup jauh karena kesehatan manusia yang tidak dapat diujicoba dan diganti-ganti dengan cara yang sembarangan diagnosa.

Bahasa pemrograman merupakan salah satu perangkat lunak yang dapat digunakan untuk membuat suatu program aplikasi yang akan membantu manusia dalam melakukan analisa, diagnosa dan prakiraan-prakiraan yang rumit. Bahasa pemrograman Visual Basic merupakan salah satu bahasa pemrograman yang berorientasi kepada obyek dan berbasis visual, sehingga program yang telah dibuat nantinya selain mampu menyelesaikan permasalahan, juga terlihat indah dalam tampilannya.

II. KAJIAN PUSTAKA

A. Kecerdasan Buatan

Rolston (1988:15) menyimpulkan dari berbagai pemahaman umum tentang kecerdasan buatan yakni : *Artificial Intelligence (AI) is the computer-solution of complex problems through the application of processes that are analogous to the human reasoning process.* Yang mana artinya lebih kurang bahwa kecerdasan buatan adalah penyelesaian dengan menggunakan komputer terhadap berbagai permasalahan kompleks melalui

proses aplikasi yang mirip dengan pola pikir manusia.

Sementara itu Negnevitsky (2002:18) menyatakan pemahaman tentang kecerdasan buatan adalah : *Artificial Intelligence is a science that has defined its goal as making machine do things that would require intelligence if done by human.* Yang mana dapat diartikan bahwa kecerdasan buatan adalah suatu ilmu pengetahuan mendefinisikan tujuan sebagai mesin yang membutuhkan kecerdasan dan pengetahuan (*knowledge-based*) jika dikerjakan oleh manusia.

Sementara itu Anita D. dan Muhammad A., (2004:1) mengutarakan pemahaman Kecerdasan Buatan (*Artificial Intelligence*) adalah *Artificial Intelligence* (AI) merupakan cabang dari ilmu komputer yang koncern dengan pengotomatisasian tingkah laku cerdas. Selain itu pemahaman definisi *Artificial Intelligence* yang lain juga dikutip oleh Anita D. dan Muhammad A., dengan tujuan untuk membuka pemahaman yang lebih luas terhadap definisi tersebut.

Adapun beberapa definisi yang dikutip tersebut dapat dilihat pada Tabel 2.1. berikut :

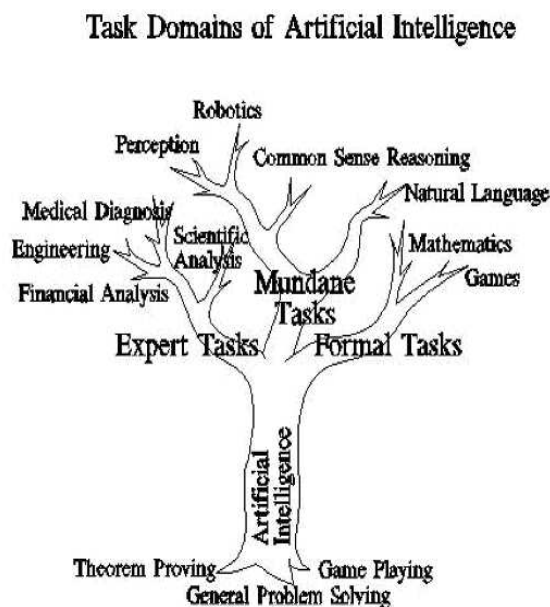
Tabel 1. Beberapa Definisi Kecerdasan Buatan (*Artificial Intelligence*)

"The existing new effort to make computers think ... machine with mind, in full and literal sense" (Haugeland, 1985)	"The study of mental faculties through the use of computational models" (Charniak and McDermott, 1985)
"The automation of activities that we associate with human thinking activities such as decision making, problem solving, learning..." (Bellman, 1978)	"The study of the computation that make it possible to perceive reason and act" (Winston, 1992)
"The art of creating machine that performs function that require intelligence when performed by people" (Kurzweil, 1990)	"A field of study that seeks to explain and emulate intelligence behavior in term of computational processes" (Schalkoff, 1990)
"The study of how to make computers do things at which, at the moment, people are better" (Rich and Knight, 1991)	"The branch of computer science that is concerned with the automation of intelligence behavior" (Luger and Stubblefield, 1995)

Berdasarkan dari pemahaman di atas, dapat disimpulkan bahwa kecerdasan buatan (*artificial intelligence*) merupakan sebuah paradigma baru

bidang ilmu komputer khususnya teknologi informasi yang mengajarkan kepada mesin komputer untuk memahami dan mempelajari sesuatu dengan kecerdasan seperti yang dimiliki manusia.

Sehingga dalam kegiatannya, mesin komputer perlu didampingi oleh manusia dalam hal masukan fakta-fakta dan teori-teori yang mempengaruhi informasi. Berikut ini disajikan ruang lingkup ilmu yang mempelajari kecerdasan buatan yang dapat dipahami seperti Gambar 2.1 berikut (Anita D. dan Muhammad A., 2004:7) :



Gambar 1. Pohon Lingkup Ilmu Artificial Intelligence

T. Sutojo dkk. (2011:1) mengungkapkan bahwa kecerdasan buatan berasal dari bahasa Inggris *Artificial Intelligence* atau disingkat AI, yaitu *Intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan.

Kecerdasan buatan yang dimaksud adalah merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia.

B. Konsep Sistem Pakar

1. Pengertian Sistem Pakar

Sistem pakar adalah system yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. (Kusumadewi, 2003 : 109)

Sistem pakar adalah sistem perangkat lunak komputer yang menggunakan ilmu, fakta, dan teknik berpikir dalam pengambilan keputusan untuk menyelesaikan masalah-masalah yang biasanya hanya dapat diselesaikan oleh tenaga ahli dalam bidang yang bersangkutan. (Andi : 2009)

Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (inference rules) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam komputer, yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu.

2. Ciri-ciri Sistem Pakar

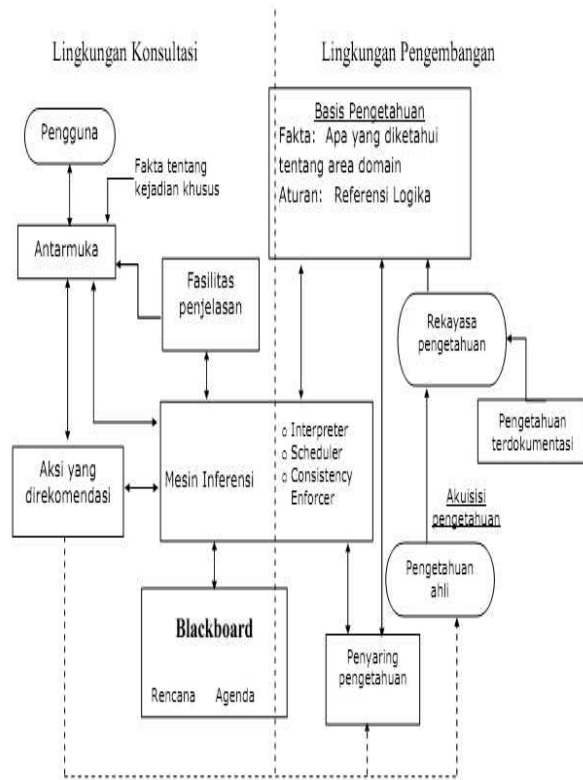
Sistem pakar yang baik harus memenuhi ciri-ciri sebagai berikut :

- a. Memiliki informasi yang handal.
- b. Mudah dimodifikasi.
- c. Dapat digunakan dalam berbagai jenis komputer.
- d. Memiliki kemampuan untuk belajar beradaptasi

3. Struktur Sistem Pakar

Sistem pakar terdiri dari 2 bagian pokok, yaitu: lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan sebagai pembangunan system pakar baik dari segi pembangunan komponen maupun basis pengetahuan.

Lingkungan konsultasi digunakan oleh seseorang yang bukan ahli untuk berkonsultasi (Gambar 2)



Gambar 2. Struktur Sistem Pakar

4. Komponen Sistem Pakar

a. Basis Pengetahuan (Knowledge Base)

Basis pengetahuan merupakan inti dari suatu sistem pakar, yaitu berupa representasi pengetahuan dari pakar. Basis pengetahuan tersusun atas fakta dan kaidah. Fakta adalah informasi tentang objek, peristiwa, atau situasi. Kaidah adalah cara untuk membangkitkan suatu fakta baru dari fakta yang sudah diketahui.

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah, tentu saja di dalam domain tertentu. Ada 2 bentuk pendekatan basis pengetahuan yang sangat umum digunakan, yaitu :

1. Penalaran Basis Aturan (Rule Base Reasoning)

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan

berbentuk : IF-THEN. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu, bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah-langkah) pencapaian solusi.

2. Penalaran Basis Kasus (Case Base Reasoning)

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip). Selain itu, bentuk ini juga digunakan apabila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan

b. Mesin Inferensi (Inference Engine)

Mesin inferensi berperan sebagai otak dari sistem pakar. Mesin inferensi berfungsi untuk memandu proses penalaran terhadap suatu kondisi, berdasarkan pada basis pengetahuan yang tersedia. Di dalam mesin inferensi terjadi proses untuk memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan dalam rangka mencapai solusi atau kesimpulan.

Dalam prosesnya, mesin inferensi menggunakan strategi penalaran dan strategi pengendalian. Strategi penalaran terdiri dari strategi penalaran pasti (Exact Reasoning) dan strategi penalaran tak pasti (Inexact Reasoning). Exact reasoning akan dilakukan jika semua data yang dibutuhkan untuk menarik suatu kesimpulan tersedia, sedangkan inexact reasoning dilakukan pada keadaan sebaliknya.

Strategi pengendalian berfungsi sebagai panduan arah dalam melakukan proses penalaran. Terdapat tiga teknik pengendalian yang sering digunakan, yaitu forward chaining, backward chaining, dan gabungan dari kedua teknik pengendalian tersebut.

Ada dua cara yang dapat dilakukan dalam melakukan inferensi, yaitu :

1. Metode Pelacakan Maju (Forward Chaining)

Metode *forward chaining* berarti menggunakan himpunan aturan kondisi-aksi atau fakta-kesimpulan. Dalam aturan fakta kesimpulan ini, fakta-fakta yang ada diuji kebenarannya untuk digunakan dalam menentukan kesimpulan apa yang ada. Metode ini cocok digunakan untuk menangani masalah pengendalian dan peramalan .

Pada sistem perantai maju, menggambarkan alur kegiatan dalam suatu sistem secara berurutan kedepan dengan pola yang sesuai dengan memori kerja.

Operasinya dimulai dengan memasukkan sekumpulan fakta yang diketahui oleh memori kerja. Kemudian diakhiri dengan pencocokan premis dengan fakta yang diketahui.

2. Metode Pelacakan Mundur (Backward Chaining)

Metode pelacakan mundur merupakan metode kebalikan dari pelacakan maju. Dalam *backward chaining*, penalaran dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut harus dicari fakta-fakta yang ada dalam basis pengetahuan.

Pengetahuan dapat dipresentasikan dalam bentuk yang sederhana atau kompleks, tergantung dari masalahnya. Secara umum representasi pengetahuan dalam sistem pakar menggunakan beberapa kaidah yaitu jaringan semantik, logika, frame, dan aturan (rule).

Dalam pembuatan aplikasi sistem pakar ini menggunakan referensi pengetahuan kaidah produksi atau aturan (rule). Aturan (rule) menyediakan secara normal untuk merepresentasikan rekomendasi, arah atau strategi. Aturan (rule) dituliskan dalam bentuk Jika-Maka (IF-THEN).

Kaidah ini menghubungkan anteseden (antecedent) dengan konsekuensi yang diakibatkannya. Berbagai struktur kaidah IF-THEN yang menghubungkan objek atau atribut adalah sebagai berikut : *Jika premis maka konklusi, Jika masukan maka keluaran, Jika kondisi maka tindakan, Jika anteseden maka konsekuensi, Jika data maka hasil, Jika tindakan maka tujuan.*

Referensi pengetahuan merupakan kombinasi sistem berdasarkan dua elemen, yaitu struktur data dan penafsiran prosedur untuk digunakan pengetahuan dalam menyimpan struktur data. Hal ini penting untuk merealisasikan kedua elemen tersebut dan dalam sistem representasi pengetahuan adalah suatu hal yang perlu.

Menurut Turban (dalam Sri Kusumadewi : 2003) konsep dasar sistem pakar mengandung keahlian, ahli, pengalihan keahlian, inferensi, aturan, dan kemampuan menjelaskan. Keahlian adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman. Ada beberapa bentuk pengetahuan yang termasuk keahlian yaitu :

1. Fakta-fakta pada lingkup permasalahan tertentu.
2. Teori-teori pada lingkup permasalahan tertentu.
3. Prosedur-prosedur dan aturan-aturan berkenaan dengan lingkup permasalahan tertentu.
4. Strategi-strategi global untuk menyelesaikan masalah.
5. Meta-knowledge (pengetahuan tentang pengetahuan).

Referensi pengetahuan dengan rule (aturan) sering disebut juga dengan sistem produksi suatu rule terdiri dari 2 bagian yaitu :

1. Antecedent yaitu bagian yang mengekspresikan situasi atau premis (pengetahuan berawal IF).
2. Konsekuensi yaitu bagian yang mengatakan sesuatu tindakan tertentu atau konklusi yang diterapkan jika situasi atau premis bernilai benar (pernyataan berawal THEN).

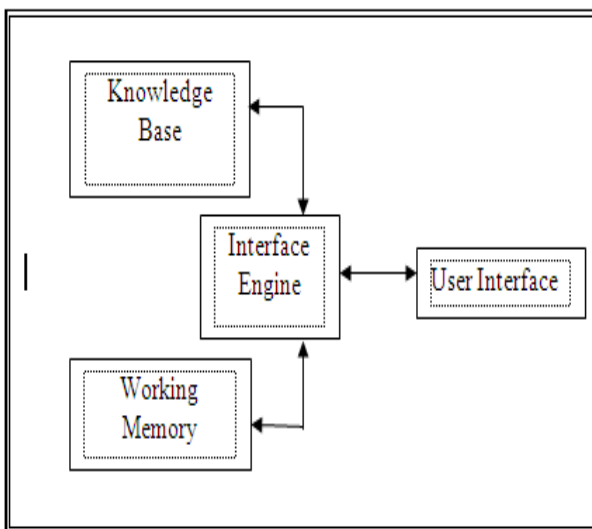
Inferensi dengan *rules* (sebagai mana juga dengan logika) dapat sangat efektif, tapi terdapat beberapa keterbatasan pada teknik-teknik tersebut. Inferensi dengan rule merupakan bagian yang menyediakan fungsi berfikir pola-pola penalaran sistem yang digunakan oleh seorang pakar tapi terdapat beberapa keterbatasan pada teknik-teknik.

c. Basis Data (Database)

Basis data terdiri atas semua fakta yang diperlukan, dimana fakta-fakta tersebut digunakan untuk memenuhi kondisi dari kaidah-kaidah dalam sistem. Basis data menyimpan semua fakta, baik fakta awal pada saat sistem mulai beroperasi, maupun fakta-fakta yang diperoleh pada saat proses penarikan kesimpulan sedang dilaksanakan. Basis data digunakan untuk menyimpan data hasil observasi dan data lain yang dibutuhkan selama pemrosesan.

d. Antarmuka Pemakai

Fasilitas ini digunakan sebagai perantara komunikasi antara pemakai dengan sistem. Antara muka pemakai (user interface) merupakan bagian penghubung antara program system pakar dengan pemakai pada bagian ini memungkinkan pengguna untuk memasukan intruksi dan informasi kedalam system pakar serta menerima penjelasan dan kesimpulan.



Gambar 3. Komponen Utama Sistem Pakar

Penjelasan dari Komponen Sistem Pakar

1. Knowledge base

Merupakan merupakan bagian dari system pakar yang berisi domain pengetahuan yang dibutuhkan untuk memahami merumuskan dan menyelesaikan masalah. *Knowledge base* terdiri dari atas 2 elemen dasar yaitu :

- Fakta, situasi masalah dan teori yang terkait
- Heuristik khusus atau rule yang langsung menggunakan pengetahuan untuk menyelesaikan masalah khusus.

2. Working memory

Merupakan bagian dari system pakar yang fakta-fakta masalah yang ditemukan dalam suatu sesi. berisi fakta masalah yang ditemukan dalam proses konsultasi.

3. inference engine

Merupakan prosesor pada system pakar yang mencocokkan fakta-fakta yang ada pada *working Memory* dengan domain pengetahuan yang terdapat *knowledge base* untuk menarik kesimpulan yang dari masalah yang dihadapi. proses berpikir.

e. Subsistem Penjelasan

Digunakan untuk melacak respon dan memberikan penjelasan tentang perilaku sistem pakar secara interaktif melalui pertanyaan:

- Mengapa suatu pertanyaan ditanyakan oleh sistem pakar?
- Bagaimana konklusi dicapai?
- Mengapa ada alternatif yang dibatalkan?
- Rencana apa yang digunakan untuk mendapatkan solusi?

f. Sistem Penyaring Pengetahuan

Sistem ini digunakan untuk mengevaluasi kinerja sistem pakar itu sendiri untuk melihat apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang

g. Keuntungan Sistem Pakar

Secara garis besar, banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain :

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian para pakar.
4. Meningkatkan output dan produktivitas.
5. Meningkatkan kualitas.
6. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
7. Mampu beroperasi dalam lingkungan yang berbahaya.
8. Memiliki kemampuan untuk mengakses pengetahuan.
9. Memiliki reabilitas.
10. Meningkatkan kapabilitas sistem komputer.
11. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
12. Sebagai media pelengkap dalam pelatihan.
13. Meningkatkan kapabilitas dalam penyelesaian masalah.
14. Menghemat waktu dalam pengambilan keputusan

h. Kelemahan Sistem Pakar

Di samping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan, antara lain :

- 1). Biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal.
- 2) Sulit dikembangkan. Hal ini tentu saja erat kaitannya dengan ketersediaan pakar di bidangnya.
- 3). Sistem Pakar tidak 100% bernilai benar.

Secara garis besar, banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain :

- 1). Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
- 2). Bisa melakukan proses secara berulang secara otomatis.

- 3). Menyimpan pengetahuan dan keahlian para pakar.
- 4). Meningkatkan output dan produktivitas.
- 5). Meningkatkan kualitas.
- 6) Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
- 7). Mampu beroperasi dalam lingkungan yang berbahaya.
- 8). Memiliki kemampuan untuk mengakses pengetahuan.
- 9). Memiliki reabilitas.
- 10). Meningkatkan kapabilitas sistem komputer.
- 11). Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
- 12). Sebagai media pelengkap dalam pelatihan.
- 13). Meningkatkan kapabilitas dalam penyelesaian masalah.
- 14). Menghemat waktu dalam pengambilan keputusan

i. Perbedaan Program Konvensional dan Sistem Pakar

Perbedaan antara pemrograman konvensional dengan sistem pakar, antara lain :

Table 2. Perbedaan Sistem Konvensional dan Sistem Pakar

Sistem konvensional	Sistem pakar
Informasi dan pemrosesnya biasanya jadi satu dengan program	Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi
Biasanya tidak bias menjelaskan mengapa suatu input data itu dibutuhkan atau bagaimana output itu diperoleh	Penjelasan adalah bagian terpenting dari sistem pakar
Pengubahan program cukup sulit dan membosankan	Pengubahan aturan dapat dilakukan dengan mudah
Sistem hanya akan beroperasi jika sistem tersebut sudah lengkap	Sistem dapat beroperasi hanya dengan beberapa aturan
Eksekusi dilakukan langkah demi langkah	Eksekusi dilakukan pada keseluruhan basis pengetahuan
Menggunakan data	Menggunakan pengetahuan
Tujuan utamanya adalah efisien	Tujuan utamanya adalah efektivitas

C. Penyakit Asma

Penyakit Asma (*asthma*) adalah suatu penyakit kronis menahun yang menyerang saluran pernapasan (*brochiale*) pada paru-paru terdapat peradangan (inflamasi) pada dinding rongga bronchiale sehingga mengakibatkan penyempitan Saluran napas yang akhirnya seseorang mengalami sesak napas. Penyakit asma paling banyak ditemukan di negara maju, terutama yang tingkat polusi udaranya tinggi baik dari asap kendaraan maupun debu padang pasir (Anonim, 1005:17).

1. Gejala Asma

Dasar kelainan asma adalah keadaan bronkus (saluran napas bagian dalam) yang hiperaktif terhadap berbagai rangsangan. Jika ada rangsangan pada bronkus yang hiperaktif akan terjadi hal-hal berikut ini :

- Otot bronkus akan mengerut atau menyempit
- Selaput lendir bronkus membengkak
- Produksi lendir menjadi banyak dan kental. Lendir yang kental ini sulit dikeluarkan (dibatukkan) sehingga penderita menjadi lebih sesak.

Keadaan bronkus yang sangat peka dan hiperaktif pada penderita asma menyebabkan saluran napas menjadi sempit, akibatnya pernapasan menjadi terganggu. Hal ini menimbulkan gejala asma yang khas, yaitu batuk, sesak napas dan *wheezing* atau mengi (napas berbunyi). Bunyi mengi dan sesak napas disebabkan oleh penyempitan saluran napas, sedangkan batuk disebabkan oleh produksi lendir yang berlebihan (gejala ini timbul jika ada suatu faktor pencetus berhubungan dengan bronkus yang hiperaktif).

Manifestasi serangan asma tidak sama pada setiap orang. Bahkan, pada satu penderita yang sama, berat dan lamanya serangan dapat berbeda dari waktu ke waktu. Beratnya serangan dapat bervariasi, mulai dari yang ringan sampai yang berat. Demikian pula dengan lamanya serangan, serangan bisa saja singkat, sebaliknya dapat pula berlangsung sampai berhari-hari. Di luar waktu serangan, biasanya penderita berada dalam keadaan sehat, seperti orang normal lainnya.

2. Klasifikasi Asma

Berdasarkan konsensus internasional dan konsensus nasional, asma diklasifikasikan menjadi 3 kelompok sebagai berikut :

- Asma episodik jarang
- Asma episodik sedang
- Asma persisten (asma berat)

Klasifikasi tersebut berdasarkan pada 8 macam penilaian, yaitu frekuensi serangan, lama serangan, intensitas serangan, keadaan di antara serangan, tidur dan aktivitas, pemeriksaan fisik di luar serangan, perlu atau tidaknya obat pengendali (obat anti inflamasi), fungsi paru di luar serangan.

Setelah diagnosis dan klasifikasi ditentukan, dokter akan merencanakan pengobatan untuk penderita.

Tabel 3. Klasifikasi Asma

No.	Parameter Klinis	Asma Episodik Jarang (Asma ringan)	Asma Episodik Sering (Asma sedang)	Asma Persisten (Asma berat)
1.	Frekuensi Serangan	Kurang dari 1 kali/bulan	Lebih dari 1 kali/bulan	Sering
2.	Lama Serangan	Kurang dari 1 minggu	1 minggu atau lebih	Hampir sepanjang tahun, tidak ada remisi
3.	Intensitas Serangan	Ringan	Berat	Lebih berat
4.	Diantara Serangan	Tanpa gejala	Sering ada gangguan	Gejala Siang dan malam
5.	Tidur dan aktivitas	Tidak terganggu	Sering terganggu	Sangat terganggu
6.	Pemeriksaan fisik di luar serangan	Normal	Mungkin terganggu	Tidak pernah terganggu
7.	Obat pengendali (anti inflamasi)	Tidak perlu	Perlu, nonsteroid	Perlu steroid
8.	Fungsi paru di luar serangan	FEF/FEV1 lebih dari 80 %	FEF/FEV1 60 - 80 %	FEF/FEV1 < 60 %

3. Pengobatan Serangan Asma

Serangan asma akut dapat dibedakan menjadi serangan asma akut, ringan, serangan akut sedang dan serangan asma akut berat. Pasien perlu mengenali beratnya serangan asma karena serangan asma yang berat dapat mengancam jiwa. Pasien perlu segera dibawa ke rumah sakit untuk mendapatkan pertolongan jika terdapat gejala-gejala di bawah ini :

1. Tidak dapat bernapas, badannya bungkuk ke depan, dan berbicara terputah-putah. Dan pada bayi, bayi tidak dapat minum, kesadaran menurun, pernapasan cepat, dangkal.
2. Pasien mengi sangat keras atau tidak terdengar, untuk lebih jelasnya, klasifikasi serangan asma dapat dilihat pada tabel 2.2. Serangan asma akut berat tidak harus didasarkan oleh asma *persisten* berat. Dapat saja suatu asma episodik jarang (derajat asma yang paling ringan) mendapat serangan asma akut berat meskipun jarang. Demikian pula tidak semua gejala atau parameter harus ada.

Tabel 4. Derajat Serangan Asma

Gejala Klinis	Serangan Ringan	Serangan Sedang	Serangan Berat
	Berjalan sudah sesak	Berbicara sudah sesak	Istirahat sudah sesak
Sesak Napas	Masih dapat berbaring	Lebih enak duduk, berbaring sesak	Duduk harus membungkuk ke depan karena sesak
Berbicara	Dapat menyelesaikan kalimat	Berbicara terputus-putus	Sukar berbicara karena sesak
Kegelisahan	Kadang-kadang gelisah	Selalu gelisah	Selalu gelisah
Frekuensi pernapasan	Meningkat	Meningkat	Sering lebih dari 30 per menit
Otot-otot bantu napas	Biasanya tidak digunakan	Biasanya digunakan	Biasanya digunakan
Bising mengi	Sedang, sering hanya akhir ekspirasi	Keras	Keras
Nadi/menit	Kurang dari 100	100/120	Lebih dari 120

Telah disebutkan sebelumnya bahwa serangan asma secara potensial dapat mengancam nyawa. Oleh karena itu, pengobatan dan penilaian keadaan penderita harus akurat dan tepat. Secara ideal hal ini harus dilakukan di rumah sakit. Meskipun demikian, ada pendapat bahwa strategi pengobatan serangan asma yang terbaik adalah pengobatan dini.

D. Sistem Komputer

Sutanta (2005:27) mengungkapkan agar komputer dapat digunakan untuk mengolah data, maka harus dibentuk sistem komputer (*computer system*). Sistem komputer terdiri atas elemen-elemen dasar berupa perangkat keras (*hardware*), perangkat lunak (*software*) dan manusia (*brainware*).

1. Perangkat Keras (Hardware)

Perangkat keras adalah peralatan sistem komputer yang secara fisik dapat dilihat/dijamah. Komponen utama perangkat keras sistem komputer adalah sebagai berikut :

- a. Unit masukan (*input*), adalah perangkat keras yang berfungsi sebagai alat untuk memasukkan data atau perintah ke dalam komputer. Unit masukan terdiri atas alat masukan langsung dan alat masukan tidak langsung. Contohnya adalah keyboard, mouse, disk drive, touch screen, light pen dan sebagainya.
- b. Unit pusat pengolah (CPU), merupakan otak sistem komputer sebagai tempat untuk mengolah instruksi program yang pada komputer ukuran mikro disebut sebagai *micro processor*.
- c. Unit keluaran (*output*), adalah perangkat keras yang berfungsi untuk menampilkan/mencetak keluaran sebagai hasil pengolahan data. Unit keluaran ini berupa *hard copy device*, *soft copy device* dan *drive device*. Contohnya printer, plotter, monitor, speaker, LCD monitor dan lain-lain.

- d. Media penyimpan (*memory*), digunakan untuk menyimpan data, informasi dan program pengolah. Memori dibedakan menjadi dua, yaitu memori internal dan memori eksternal. Memori internal berupa RAM (*Random Access Memory*) yang berfungsi untuk menyimpan program yang diolah sementara waktu dan ROM (*Read Only Memory*) yaitu memori yang hanya bisa dibaca dan berguna sebagai penyedia informasi pada saat komputer pertama kali dinyalakan.

2. Perangkat Lunak (Software)

Perangkat lunak komputer merupakan serangkaian instruksi dengan aturan tertentu yang mengatur operasi perangkat keras, Perangkat lunak terdiri atas 3 kelompok, yakni :

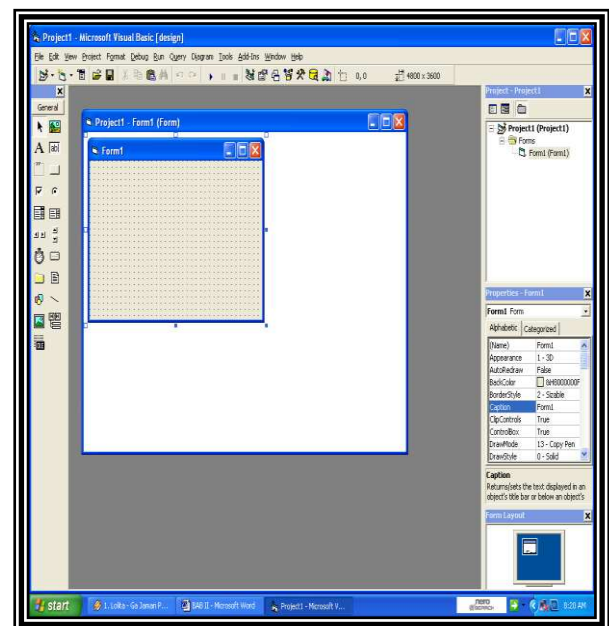
- a. Sistem Operasi (*Operating System*), adalah perangkat lunak yang digunakan untuk mengendalikan *resources* selama proses berlangsung. Fungsi sistem operasi adalah mengalokasikan memori dan pekerjaan dan penjadwalan pekerjaan. Contoh sistem operasi adalah DOS, Unix, Novell, OS/2, Windows.
 - b. Bahasa Pemrograman (*Programming Language*), adalah merupakan perangkat lunak yang digunakan untuk pengembangan program aplikasi. Level/generasi bahasa pemrograman komputer dikenal dengan 4 GL, yaitu: bahasa mesin (*machine language*), bahasa rakitan (*assembly language*), bahasa pemrograman (*programming language*) dan aplikasi (*application*).
 - c. Aplikasi (*Application*), adalah merupakan perangkat lunak yang dikembangkan untuk digunakan pada aplikasi tertentu. Contoh GL (*General Ledger*), MYOB, Payroll dan sebagainya.
- ## 3. Perangkat Cerdas (Manusia)
- Perangkat Cerdas (*Brainware*) adalah merupakan personal yang terlibat dalam sistem

komputer, yaitu *operator*, *programmer*, *system analyst*.

E. Bahasa Pemrograman Visual Basic

Pemrograman Visual Basic 6.0 adalah salah satu bahasa pemrograman komputer. Menurut Hidayat Sapar (2010 : 1) Bahasa pemrograman adalah perintah-perintah atau instruksi-instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Visual Basic selain disebut Bahasa pemrograman juga sering disebut sebagai sarana (*tool*) untuk menghasilkan program-program aplikasi berbasis windows.

Hal ini lebih mudah lagi setelah hadirnya Microsoft Visual Basic, yang dibangun dari ide untuk membuat bahasa yang sederhana dan mudah dalam pembuatan scriptnya (*simple scripting language*) untuk graphic user interface yang dikembangkan dalam sistem operasi Microsoft Windows.



Gambar 4. Halaman Menu Program Visual Basic

Adapun Komponen-komponen yang terdapat dalam Visual Basic dan dapat digunakan untuk keperluan perancangan tatap muka (*interface*) suatu program aplikasi adalah dapat dijelaskan sebagai berikut :

1. Baris Menu

Baris menu (*Menu Bar*) merupakan fasilitas standar yang disediakan oleh program-program yang bekerja di bawah fasilitas windows. Baris menu ini dapat dipanggil dengan cara, melakukan penekanan tombol Alt pada keyboard disertai dengan melakukan penekanan huruf yang bergaris bawah atau dapat dilakukan dengan langsung mengklik pada menu dengan menggunakan mouse.



Gambar 5. Baris Menu

2. Tools Bar

Toolbars ini digunakan dengan cara menunjuk panah/pointer pada salah satu tombol yang tersedia dalam menu tersebut yang diwakili oleh gambar (icon) yang telah tersedia



Gambar 6. Tools Bar

3. Jendela Tool Box

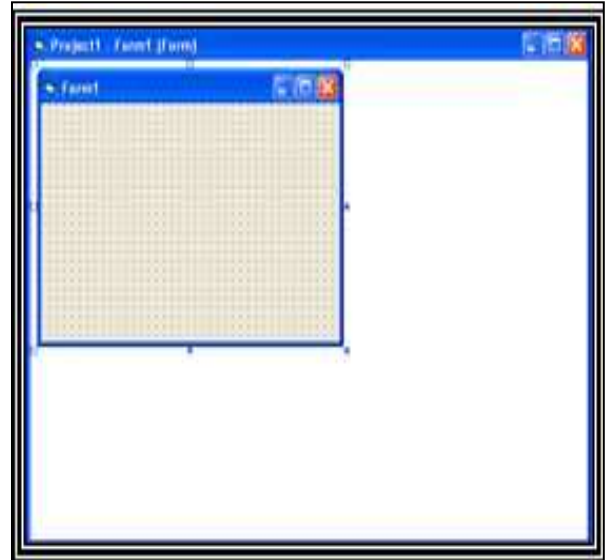
Pada jendela Toolbox ini berisikan tentang perlengkapan dalam perancangan Visual Basic yang diinginkan.



Gambar 7. Jendela Tool Box

4. Jendela Form

Jendela Form merupakan jendela aplikasi tempat pembuatan program yang akan ditampilkan



Gambar 8. Jendela Form

5. Jendela Project

Jendela Project merupakan jendela informasi tentang project yang sedang dibuka beserta semua sub program serta segala pendukungnya.



Gambar 9. Jendela Project

6. Jendela Properties

Jendela Properties merupakan jendela yang berisikan berbagai macam string yang dapat dirubah berkaitan dengan program yang dirancang.



Gambar 10. Jendela Properties

Sebelum, merancang sebuah file terlebih dahulu kita mendefinisikan struktur file dengan menggunakan *Project Module* yang ada dalam Visual Basic. Pada project Module kita deklarasikan semua struktur file kedalam kode type yang dimengerti oleh Visual Basic. Pendeklarasian Struktur file harus disesuaikan supaya tidak terjadi pengulangan. Oleh sebab itu untuk menghindari hal demikian maka terlebih dahulu dibuat dalam bentuk table.

Setelah semua file telah dideklarasikan maka kita akan memulai membuat user interface kedalam form yang telah tersedia, dalam membuat suatu interface harus diperhatikan letak dan susunan sehingga dapat dimengerti oleh User.

Setelah itu kita dapat memulai perancangan dan pembuatan suatu aplikasi bahasa pemrograman. Dan jika selesai merancang kita dapat keluar dari program Visual Basic dengan cara mengklik icon *close* atau melalui Alt+F4 atau melalui klik File Exit.

F. Konsep Perancangan Database (Database Management System)

Nugroho (2005:67) mengungkapkan bahwa Data Base (basis data) adalah sebuah cara mendokumentasikan berbagai macam data yang kemudian dimanajemen dengan sebuah system untuk kemudian disimpan dalam sebuah media penyimpanan. Adapun ciri-ciri basis data diantaranya adalah :

1. Efisiensi meliputi kecepatan, ukuran, dan ketepatan
2. Data dalam jumlah besar.
3. Berbagi Pakai (dipakai bersama sama/*Sharebility*).
4. Mengurangi bahkan menghilangkan terjadinya duplikasi dan ketidak-konsistenan data

Menurut Sinarmata dan Paryudi (2006:8) Sistem manajemen basis data (DBMS) adalah perangkat lunak yang didesain untuk membantu memelihara dan memanfaatkan kumpulan data yang besar. Kebutuhan akan sistem termasuk pula penggunaannya yang berkembang pesat.

Alternatif penggunaan DBMS adalah menyimpan data dalam file dan menulis kode aplikasi tertentu untuk mengaturnya. Penggunaan DBMS memiliki beberapa manfaat atau keuntungan penting, diantara keuntungannya adalah sebagai berikut :

- a. Mengurangi Pengulangan Data : Apabila dibandingkan dengan file-file komputer yang disimpan terpisah disetiap aplikasi komputer, DBMS mengurangi jumlah total file dengan menghapus data yang terduplikasi di berbagai file. Data terduplikasi selebihnya dapat ditempatkan dalam satu file.
- b. Mencapai Independensi Data : Spesifikasi data disimpan dalam skema pada tiap program aplikasi. Perubahan dapat dibuat pada struktur data tanpa mempengaruhi program yang mengakses data.
- c. Mengintegrasikan Data Beberapa File : Saat file dibentuk sehingga menyediakan kaitan logis, maka organisasi fisik bukan merupakan

kendala. Organisasi logis, pandangan pengguna, dan program aplikasi tidak harus tercermin pada media penyimpanan fisik.

- d. Mengambil Data dan Informasi dengan cepat : Hubungan-hubungan logis, bahwa manipulasi data, serta bahasa query memungkinkan pengguna mengambil data dalam hitungan detik atau menit.
- e. Meningkatkan Keamanan : DBMS mainframe maupun komputer mikro dapat menyertakan beberapa lapis keamanan seperti kata sandi (password), direktori pemakai, dan bahasa sandi (encryption) sehingga data yang dikelola akan lebih aman

Sistem manajemen basis data atau juga bisa disebut Database Management System (DBMS) yaitu berupa suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna.

DBMS juga merupakan perangkat lunak yang dirancang untuk dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah yang besar dan dirancang untuk dapat melakukan manipulasi data secara lebih mudah.

Dalam penyimpanan dengan DBMS, terdapat beberapa manfaat yang terdapat pada DBMS diantaranya :

1. Independensi. Perubahan struktur database dimungkinkan terjadi tanpa harus mengubah aplikasi yang mengaksesnya sehingga pembuatan antarmuka ke dalam data akan lebih mudah dengan penggunaan DBMS.
2. Penyimpanan dalam bentuk DBMS cukup besar, sangat jauh berbeda dengan disimpan dalam bentuk flat file. Disamping memiliki unjuk kerja yang lebih baik, juga akan didapatkan efisiensi penggunaan media penyimpanan dan memori.
3. Independensi. Perubahan struktur database dimungkinkan terjadi tanpa harus mengubah aplikasi yang mengaksesnya sehingga

pembuatan antarmuka ke dalam data akan lebih mudah dengan penggunaan DBMS.

4. Integritas data lebih terjamin dengan penggunaan DBMS. Masalah redudansi atau kejadian berulangnya data atau kumpulan data yang sama dalam sebuah database yang mengakibatkan pemborosan media penyimpanan sering terjadi dalam DBMS.
5. Sekuritas. DBMS memiliki sistem keamanan yang lebih fleksibel daripada pengamanan pada file sistem operasi. Keamanan dalam DBMS akan memberikan keluwesan dalam pemberian hak akses kepada pengguna.
6. Sentralisasi. Data yang terpusat akan mempermudah pengelolaan database. kemudahan di dalam melakukan bagi pakai dengan DBMS dan juga kekonsistenan data yang diakses secara bersama-sama akan lebih terjamin dari pada data disimpan dalam bentuk file atau worksheet yang tersebar.

Berikut ini fungsi-fungsi di dalam Database Management System :

- a. Data Definition : Sebuah DBMS harus bisa mengolah dari pendefinisian data
- b. Data Manipulation : Sebuah DBMS harus bisa menyelesaikan permintaan-permintaan dari pemakaian untuk mengakses data.
- c. Data Security dan Integrity : Sebuah DBMS harus dapat memeriksa security dan integrity data yang didefinisikan oleh DBA.
- d. Data Recovery dan Concurrency: sebuah DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan disk, dan sebagainya dan juga harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat yang bersamaan.
- e. Data Dictionary : Sebuah DBMS harus dapat menyediakan data dictionary

- f. **Perfomance** : Sebuah DBMS harus dapat menangani unjuk kerja dari semua fungsi seefisien mungkin.

DBMS memiliki keuntungan seperti berikut:

- Independensi data** : DBMS menyediakan pendekatan yang membuat perubahan dalam data tidak membuat program harus diubah
- Pengaksesan yang efisien terhadap data** : DBMS menyediakan berbagai teknik yang canggih sehingga penyimpanan dan pengambilan data dilakukan secara efisien.
- Keamanan dan integritas data** : Karena data dikendalikan oleh DBMS, DBMS dapat melakukan kendala integritas terhadap data. Segala sesuatu yang tidak sesuai dengan definisi suatu field dan kekangan yang melekat pada field akan ditolak. Sebagai contoh, jika field Jenis_Kelamin dinyatakan berupa P atau W, maka penyimpanan L ke field tersebut dengan sendirinya akan ditolak oleh DBMS.
- Administrasi data** : Jika sejumlah pemakai berbagi data, pemusatan administrasi dapat meningkatkan perbaikan yang sangat berarti. Dengan cara seperti ini, duplikasi atau redundansi data dapat diminimalkan.
- Akses bersamaan dan pemulihan terhadap** Waktu pengembangan aplikasi terpendek. DBMS menawarkan banyak fasilitas yang memudahkan dalam menyusun aplikasi sehingga waktu pengembangan aplikasi dapat diperpendek.

G. Entity Relationship Diagram (ERD)





Menurut Simarmata dan Paryudi (2006:59) Entity Relationship (ER) data model didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut dengan entitas dan hubungan antar objek. Entitas adalah sesuatu atau objek dalam dunia nyata yang dapat dibedakan dari objek lain. Sebagai contoh masing-masing mahasiswa adalah entitas dan mata kuliah bisa juga dianggap sebagai entitas.

Model keterhubungan entitas yang berisikan komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut untuk mempresentasikan seluruh fakta yang digambarkan dengan sistematis dalam bentuk diagram.

Komponen dasar model merupakan diagram *entity-relationship* yang digunakan untuk menyajikan objek data secara visual. Kegunaan model ER dalam perancangan tersebut adalah :

- Mampu memetakan model relasional dengan baik. Pembangunan yang digunakan didalam model ER dengan mudah diubah ke dalam tabel relasional.
- Sederhana dan mudah dipahami hanya dengan sedikit pelatihan. Oleh karena itu, model bisa digunakan oleh perancang basis data untuk mengomunikasikan perancangan kepada pengguna akhir.
- Sebagai tambahan, model bisa digunakan sebagai suatu rencana perancangan oleh pengembang basis data untuk menerapkan suatu model data dalam perangkat lunak manajemen basis data spesifik

Dalam pembuatan diagram ER terdapat beberapa simbol yang harus dimengerti fungsi dan prinsip dari simbol diantaranya adalah :

No	Simbol	Nama	Keterangan
1		Entitas	Merupakan suatu objek yang didefinisikan dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang dibuat. Entitas digambarkan dengan simbol yang disebut dengan <i>rectangle box</i>
2		Atribut	Memiliki elemen yang disebut dengan atribut yang berfungsi untuk menjelaskan karakter dari entitas dan sebagai key digaris bawah
3		Line atau garis	Berfungsi untuk menghubungkan atribut dengan yang lainnya. Hubungan ini <i>relationship</i> relasi
4		Hubungan entitas	Dapat menghubungkan satu dengan yang lainnya. Hubungan ini dinamakan <i>relationship</i> relasi

Gambar 11. Simbol ERD


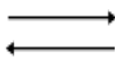


Dalam membuat diagram ER, terdapat metodologi ERD diantaranya adalah sebagai berikut :

Menentukan Entitas	Menentukan peran, kejadian, lokasi, hal nyata dan konsep dimana pengguna akan menyimpan data
Menentukan Relasi	Tentukan hubungan antara pasangan entitas menggunakan matriks relasi
Gambar ERD sementara	Entitas digambarkan dengan kotak dan relasi dengan garis yang menghubungkan entitas.
Isi Kardinalitas	Tentukan jumlah kejadian dari satu entitas untuk sebuah kejadian pada entitas yang berhubungan
Tentukan Kunci Utama	Tentukan atribut yang mengidentifikasi satu dan satu kejadian pada masing-masing entitas.
Gambar ERD berdasarkan kunci	Hilangkan relasi many-to-many dan masukkan primary dan kunci tamu pada masing-masing entitas.
Menentukan Atribut	Tuliskan field-field yang diperlukan oleh sistem
Pemetaan Atribut	Pasangkan Atribut dengan satu entitas yang sesuai pada masing-masing atribut
Gambar ERD dengan Atribut	Aturlah ERD dari langkah 6 dengan menambahkan entitas atau relasi yang ditemukan pada langkah 8.
Periksa Hasil	Apakah ERD sudah menggambarkan sistem yang akan dibangun.

H. Data FlowDiagram (DFD)

DFD (*data flow diagram*) merupakan diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem. (Jogianto, 2005 : 700).

DFD (*data flow diagram*) sering digunakan untuk menggambarkan suatu system yang telah ada atau system baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir (misalnya lewat telepon) atau lingkungan fisik tersebut akan disimpan. Berikut ini adalah symbol-simbol dalam merancang DFD (*data flow diagram*).

No	Simbol	Nama	Keterangan
1		External entity (kesatuan luar)	Digunakan untuk memisahkan suatu system dengan lingkungan luarnya.
2		Data flow (arus data)	Menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem
3		Process (proses)	Menunjukkan kegiatan perubahan alir data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses
4		Data store (simpanan data)	Merupakan media penyimpanan data yang akan digunakan untuk menyimpan hasil aliran data dari sebuah sumber pemrosesan data sampai dengan sebuah proses data

Gambar 12. Simbol Data Flow Diagram (DFD)

Dalam DFD dibagi lagi menjadi beberapa proses didalamnya untuk memudahkan user atau pemakai yaitu :

a. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. Ia akan memberikan gambaran tentang keseluruhan sistem.

b. Diagram Nol

Diagram nol adalah diagram yang menggambarkan proses dari dataflow diagram. Diagram nol memberikan

pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal entity. Pada level ini sudah dimungkinkan adanya/digambarkannya data store yang digunakan.









a. Diagram Detail

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram zero atau diagram level di atasnya

I. Flowchart

Flowchart atau Bagan alir adalah bagan (*Chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. (Jogianto, 2005 : 795).

Bagan alir digunakan untuk alat bantu komunikasi dan untuk dokumentasi. Berikut ini simbol-simbol yang digunakan dalam bagan alir :

	Symbol yang digunakan untuk mewakili proses
	Symbol alir data yang digunakan untuk menunjukan arus dari suatu proses
	Symbol alir yang digunakan untuk menunjukan sambungan dari alir yang terputus di halaman yang sama
	Symbol keputusan digunakan untuk menunjukan awal akhir dari suatu proses
	Symbol titik terminal digunakan untuk menunjukan awal dan akhir dari suatu proses
	Digunakan untuk menunjukan arus proses yang terputus dengan sambungan yang ada di halaman yang lain
	Symbol persiapan digunakan untuk memberikan nilai awal suatu besaran
	Symbol input/output mewakili data input maupun data output

Gambar 13. Flowchart

III. ANALISIS DAN PERANCANGAN

a. Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode Pengembangan, dimana langkah-langkah yang dilakukan adalah:

1. Meneliti dan mengumpulkan data-data yang diperlukan.
2. Membuat rancangan input dan output.
3. Membuat rancangan dalam bahasa pemrograman Visual Basic 6.0
4. Studi pustaka, yaitu mengumpulkan bahan pustaka yang sesuai dengan bidang ilmu yang diteliti, baik dari perpustakaan, buku-buku maupun dari internet.

b. Perangkat Software dan Hardware yang digunakan

Software yang digunakan dalam penelitian ini adalah : Sistem Operasi yang digunakan adalah MicroSoft Windows XP, Paket Aplikasi MicroSoft Office 2007, MicroSoft Visual Basic 6.0 sebagai perangkat lunak utama yang akan digunakan untuk merancang program

Hardware yang digunakan dalam penelitian ini dengan spesifikasi : Pada proses perancangan program, perangkat keras (*hardware*) yang digunakan dengan spesifikasi sebagai berikut : CPU Prosesor Intel Core 2 Duo merk Dell, Ram 2 GB, Hardisk 160 GB, Monitor DELL 17 n , Camera 1.3 pixel, DVD \pm RW, Internal Speaker 2 x 2 w, Printer Cannon IP 2700

c. Analisa Sistem Yang Berjalan

Sebelum melakukan analisa lebih terhadap rencana pengembangan suatu sistem baru yang diterapkan penulis melakukan tinjauan terhadap sistem lama, supaya dapat mengetahui kelemahan-kelemahan sistem yang dipakai pada Puskesmas Lingkar Timur Kecamatan Gading Cempaka Kota Bengkulu.

d. Konsep Perancangan Sistem Baru

Dalam merancang sistem baru ini, penulis akan merancang sebuah perangkat lunak atau program untuk memudahkan pekerjaan dan memberikan informasi tentang kondisi pelayanan kesehatan yang ada di wilayah Kelurahan Panorama Kecamatan Gading Cempaka Kota Bengkulu.

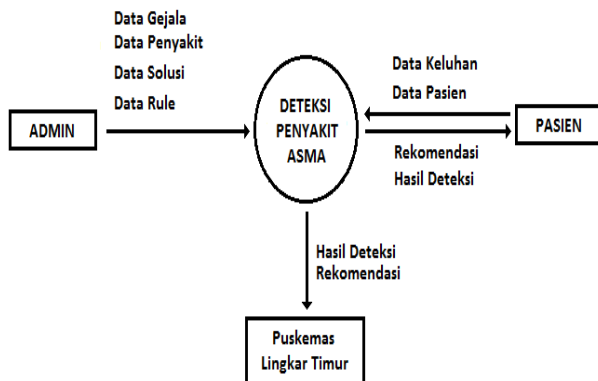
Selanjutnya proses pembuatan program ini akan didukung langkah-langkah kerja yang disebut *Data Flow Diagram*, digunakan untuk menggambar sistem yang berjalan yang digambarkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data disimpan dan menggunakan ERD untuk menggambarkan kebutuhan database.

e. Data Flow Diagram

Diagram alir data perlu dijelaskan dengan maksud untuk mengetahui status aliran dari asal data diinputkan, diproses, disimpan kemudian ditampilkan dalam bentuk data baru maupun informasi baru.

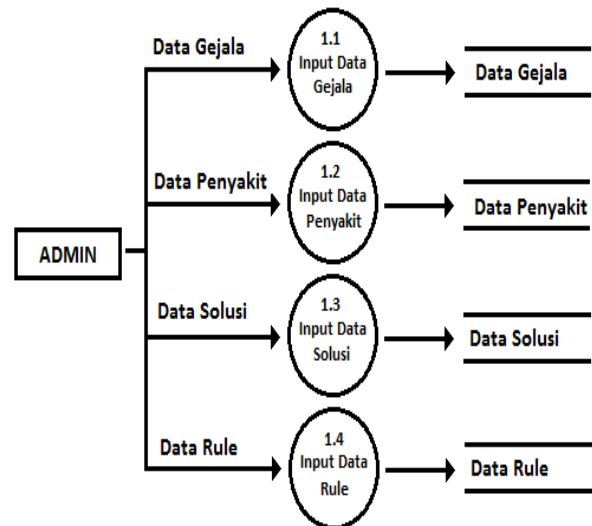
Diagram Konteks

Diagram konteks adalah merupakan gambaran secara grafis hubungan antara kesatuan luar dengan sistem yang ada. Kesatuan luar yang terlibat dalam program ini adalah Admin (*administrator*) merupakan kesatuan luar yang paling berperan terhadap entry data-data utama.

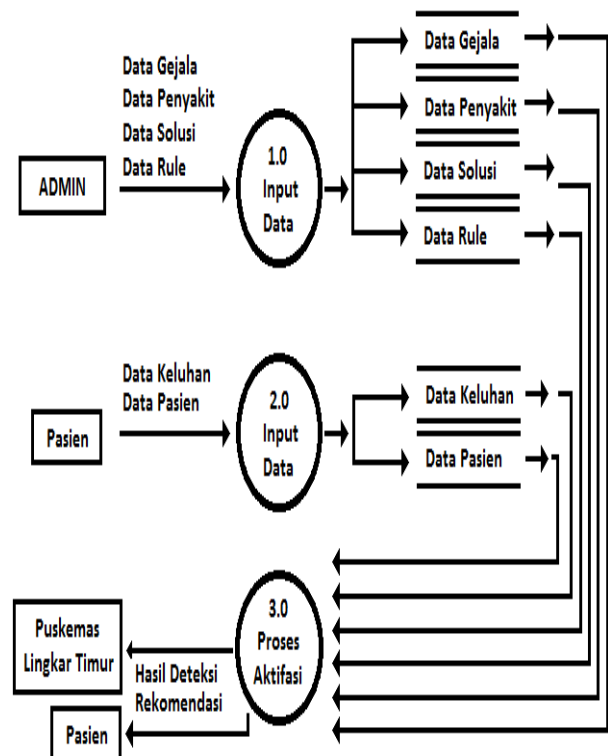


Gambar 14 Diagram Konteks

Kesatuan luar berikutnya adalah Pasien sebagai pengguna jasa pelayanan kesehatan. Selanjutnya kesatuan luar berikutnya adalah Puskesmas Lingkar Timur selaku pengelola. Adapun bentuk diagram konteks tersebut dapat dilihat pada gambar 14.



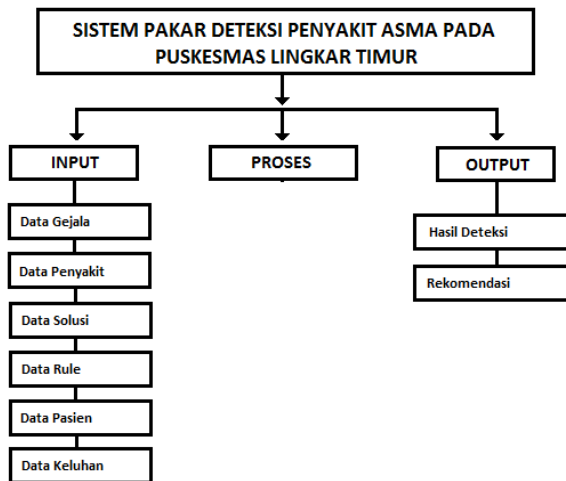
Gambar 15 Diagram Input Data



Gambar 15. Diagram Proses Detail

f. Perancangan Struktur Menu

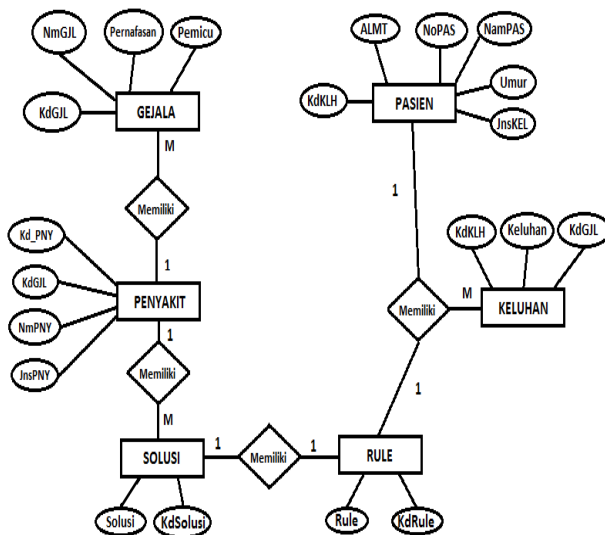
Dalam perancangan struktur menu menggambarkan dalam aplikasi system pakar menu terbagi menjadi dua menu yaitu menu user dan menu admin. Sebagai contoh struktur menu admin pada gambar 16



Gambar 16 Struktur Menu Admin

g. Entity Relationship Diagram

Keterhubungan data yang terdapat dalam database aplikasi system pakar ini dapat tergambarkan dalam diagram ERD



Gambar 17. Entity Relatinship Diagram

h. Perancangan Interface Login

Dalam membangun aplikasi, tahapan perancangan interface berguna untuk mengimplementasikan ke dalam bahasa pemrograman visual basic 6.0. dengan interface utama yang akan menghubungkan ke sub menu user / pemakai dan pakar :

Gambar 18. Rancangan Interface Login

i. Perancangan Database

Dalam membangun aplikasi system pakar ini dibutuhkan 7 database yang dijadikan sebagai storage penyimpanan data untuk masing-masing yang akan diinputkan nantinya:

1. Database Admin

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	NamAdmin	Text	25	Nama Admin
2.	Password	Text	5	Passwotd

2. Database Pasien

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	NoPAS *	Text	5	Nomor Pasien
2.	NmPAS	Text	25	Nama Pasien
3.	Alamat	Text	40	Alamat Pasien
4.	Umur	Byte	2	Umur Pasien
5.	JnsKEL	Text	9	Jenis Kelamin
6.	KdKLH **	Text	2	Kode Keluhan

3. Database Keluhan

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	KdKLH *	Text	2	Kode Keluhan
2.	Keluhan	Text	25	Keluhan
3.	KdGJL **	Text	3	Kode Gejala

4. Database Gejala

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	KdGJL *	Text	3	Kode Gejala
2.	NmGJL	Text	25	Nama Gejala

5. Database Penyakit

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	KdPNY *	Text	5	Kode Penyakit
2.	NmPenyakit	Text	25	Nama Penyakit

6. Database Solusi

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	KdSolusi *	Text	2	Kode Solusi
2.	Solusi	Memo	-	Solusi
3.	KdPeny **	Text	5	Kode Penyakit

7. Database Rule

No.	Nama Field	Tipe	Ukuran	Keterangan
1.	KdRule *	Text	2	Kode Rule
2.	Rule	Memo	-	Rule
3.	KdSolusi **	Text	2	Kode Solusi

j. Perancangan Input Data

Berdasarkan dengan database yang dibutuhkan aplikasi system pakar, perancangan input data terdiri dari :

1) Input Data Admin

Gambar 19 Rancangan Input Data Admin

2) Input Data Pasien

Gambar 20 Rancangan Input Data Pasien

3) Input Data Keluhan

DATA KELUHAN

Kode Keluhan

Uraian Keluhan

Kode Gejala ▼

Nama Gejala

Kode Keluhan	Uraian Keluhan	Kode Gejala	Nama Gejala
x-5-x	xx - 20 - xx	x-5-x	xx - 20 - xx

Tambah Koreksi Hapus Simpan Keluar

Gambar 20. Rancangan Input Data Keluhan

4) Input Data Gejala

DATA KELUHAN

Kode Keluhan

Uraian Keluhan

Kode Gejala ▼

Nama Gejala

Kode Keluhan	Uraian Keluhan	Kode Gejala	Nama Gejala
x-5-x	xx - 20 - xx	x-5-x	xx - 20 - xx

Tambah Koreksi Hapus Simpan Keluar

Gambar 21. Rancangan Input Data Gejala

5) Input Data Penyakit

DATA PENYAKIT

Kode Penyakit

Nama Penyakit

Jenis Penyakit

Kode Gejala ▼

Nama Gejala

Kode Penyakit	Nama Penyakit	Jenis	Kode Gejala	Nama Gejala
x-5-x	xx-20-xx	xx-15-xx	xxxx	xx-15-xx

Tambah Koreksi Hapus Simpan Keluar

Gambar 22. Rancangan Input Data Penyakit

6) Input Data Solusi

DATA SOLUSI

Kode Solusi

Solusi

Kode Penyakit ▼

Nama Penyakit

Kode Solusi	Solusi	Kode Penyakit	Nama Penyakit
x-5-x	xx-20-xx	xxxx	xx-15-xx

Tambah Koreksi Hapus Simpan Keluar

Gambar 23. Rancangan Input Data Solusi

7) Input Data Rule

Gambar 24. Rancangan Input Data Rule

k. Perancangan Informasi / Hasil Diagnosa

Berdasarkan dengan hasil rule dan diagnose pasien maka system akan menghasilkan output atau informasi

Gambar 25. Hasil Diagnosa

1. Rencana Pengujian Sistem

Sebelum program diterapkan atau di implementasikan maka program harus bebas terlebih dahulu dari kesalahan ataupun bug. Setelah program bebas dari kesalahan, program dites dengan memasukkan data untuk diolah.

Hasil program yang sesuai dengan desainnya akan menghasilkan sistem yang sesuai dengan kebutuhan pemakai.

Sommerville (2003:57) menerangkan bahwa suatu perangkat lunak perlu dilakukan verifikasi dan validasi (V & V) yaitu untuk menunjukkan bahwa sistem sesuai dengan spesifikasinya dan bahwa sistem memenuhi harapan pengguna.

Validasi melibatkan proses pemeriksaan, seperti inspeksi dan peninjauan, pada setiap tahap proses perangkat lunak dari definisi persyaratan user sampai pengembangan program.

IV. HASIL DAN PEMBAHASAN

1. I

Berdasarkan dari tahapan dalam penelitian, dari tahapan perancangan system dan implementasi system Aplikasi Sistem Pakar Deteksi Penyakit Asma Pada Puskesmas Lingkar Timur yang telah dirancang sesuai dengan proposal, diberi nama file SistemPakar.

Dimana program ini dapat di aktifkan melalui file aplikasi yang berbentuk Exe. Sedangkan source programnya baru dapat diaktifkan apabila pada sistem komputer telah terpasang Microsoft Visual Basic 6.0.

a. Tampilan Login Admin

Gambar 26. Tampilan Halaman Login

b. Halaman Login Administrator



Gambar 27. Halaman Login administrator

Berdasarkan gambar 27 diatas, Setelah benar memasukan nama administator dan password pada login program, maka selanjutnya program akan menampilkan tampilan menu utama seperti gambar 28 berikut:



Gambar 28 Halaman Menu Administrator

Dari halaman menu administrator terdiri dari empat sub menu diantaranya adalah sub menu input data, sub menu laporan, sub menu bantuan dan sub menu keluar.

Pada sub menu input data terdapat lima sub menu input data diantaranya adalah sub menu input data tambah data admin, sub menu input data gejala, sub menu input data penyakit, sub menu input data solusi dan sub menu input data rule.



Gambar 29. Halaman Sub menu Input Data

Dari menu halaman administrator terdapat menu laporan yang terdiri dari tiga laporan diantaranya adalah daftar gejala, daftar penanggulangan dan rekapitulasi konsultasi pasien.



Gambar 30. Tampilan Menu Sub Menu Laporan

c. Tampilan Input Data

Sub menu Input Data disediakan untuk melayani pengelolaan terhadap data yang akan dikerjakan. Dimana sub menu data ini memiliki 5 (lima) macam pilihan, yakni Sub-sub menu Input Data Tambah Admin, Gejala, Penyakit, Solusi dan Rule, berikut ini akan dibahas masing-masing pilihan, yaitu :

1. Input data admin

Berfungsi untuk menambah data admin, dari Menu Utama diclick Input Data dan dilanjutkan dengan mengklik Tambah Admin.

Gambar 31. Input Data Admin

2. Input Data Gejala

Untuk melakukan penambahan atau pengisian data baru, dari Input data Gejala, setelah tombol Tambah diklik, maka secara otomatis kode gejala akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif.

Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data Gejala.

Setelah pengisian kode Gejala dianggap benar, selanjutnya program meminta pengguna menekan tombol Enter pada keyboard untuk melanjutkan pada pengisian uraian gejala

Kode Gejala	Uraian Gejala
G07	Intensitas serang
G08	Intensitas serang
G09	Intensitas serang
G10	Diantara serang
G11	Diantara serang
G12	Diantara serang
G13	Tidur dan aktifitas
G14	Tidur dan aktifitas
G15	Tidur dan aktifitas

Gambar 32. Input Data Gejala

a. Tambah Data

Untuk menambah data administrator dapat dilakukan dengan cara mengisi user name dan password kemudian tombol simpan yang sebelumnya tidak aktif menjadi aktif.

Gambar 33. Tampilan Tambah Data Gejala

Setelah tombol Tambah diklik, maka secara otomatis kode gejala akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif.

Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data Gejala.

Setelah pengisian kode Gejala dianggap benar, selanjutnya program meminta pengguna menekan tombol Enter pada keyboard untuk melanjutkan pada pengisian uraian gejala. Selanjutnya pengguna diminta menekan tombol Enter untuk mengaktifkan tombol Simpan

b. Koreksi Data

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data Gejala seperti pada gambar 4.12 di atas diklik tombol Koreksi maka program akan menampilkan.

Kode Gejala	Uraian Gejala
G07	Intensitas serang
G08	Intensitas serang
G09	Intensitas serang
G10	Diantara serang
G11	Diantara serang
G12	Diantara serang
G13	Tidur dan aktifitas
G14	Tidur dan aktifitas
G15	Tidur dan aktifitas

Gambar 34. Tampilan Koreksi Data Gejala

Berdasarkan gambar koreksi gejala di atas, terlihat bahwa untuk melakukan perbaikan terhadap data dilakukan dengan cara mengetik kode Gejala yang ingin di perbaiki.

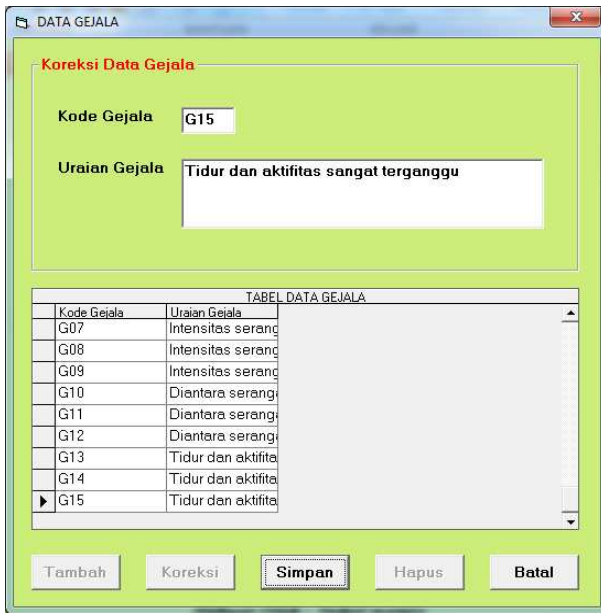
Dalam hal ini akan dilakukan perbaikan terhadap data dengan kode gejala G15, kemudian program akan membandingkan kode tersebut dengan data kode Gejala pada database, jika ditemukan maka ditampilkan isi data lengkapnya.

Apabila kode gejala yang dipilih tidak ditemukan, program kemudian menampilkan pesan data tidak ditemukan, sedangkan apabila ditemukan, program menampilkannya ke layar sebagaimana terlihat pada tampilan gambar dibawah ini:

Gambar 35. Tampilan Hapus Data Gejala

c. Simpan Data

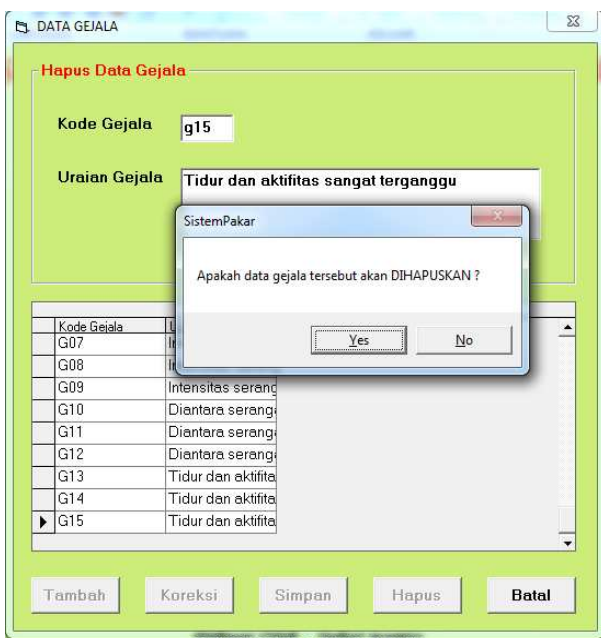
Setelah tombol Simpan pada gambar diklik, maka data terbaru dengan kode penggugat G15 telah disimpan dan program kemudian kembali seperti tampilan setelah melakukan penyimpanan terhadap data yang telah diperbaiki.



Gambar 36. Tampilan Simpan Data Gejala

d. Hapus Data

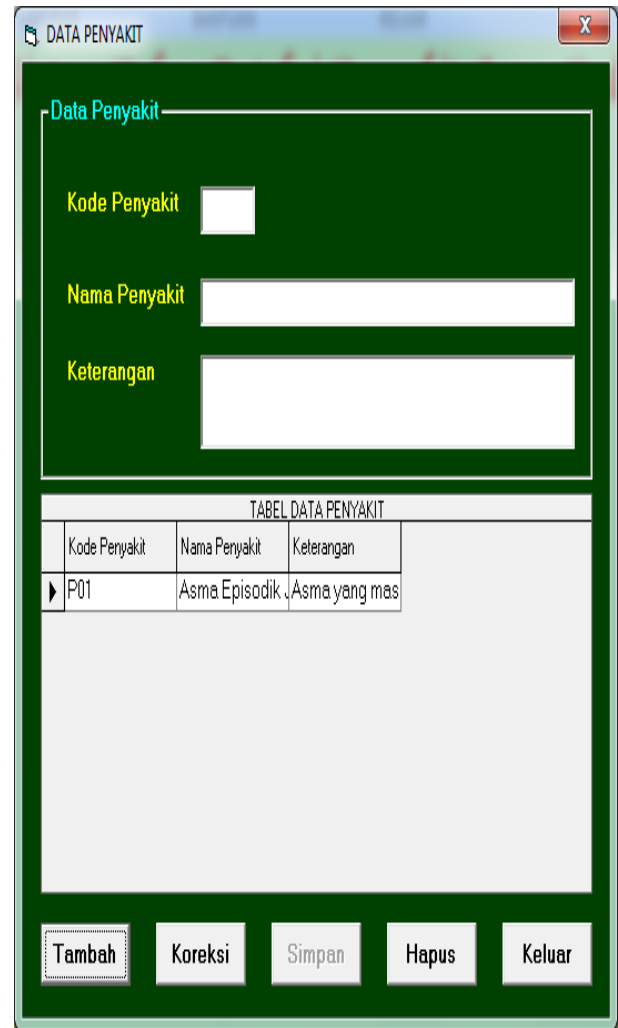
Berfungsi untuk dapat melakukan penghapusan terhadap data yang telah tersimpan dalam database. Apabila tombol Hapus diklik maka program selanjutnya akan memberikan tampilan.



Gambar 37. Tampilan Hapus Data Gejala

apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan kode gejala G15 yang telah dipilih, baik pada data Gejala maupun pada databasenya

3. Input Data Penyakit



Gambar 38. Tampilan input Data Penyakit

Untuk menampilkan Input Data Penyakit, dari Menu Utama diklik Input Data selanjutnya dengan meng-klik Penyakit, maka program akan memberikan tampilan program

Input data Penyakit berfungsi sebagai tatap muka (*interface*) guna melakukan penambahan data, perbaikan data, dan penghapusan data penyakit.

a. Tambah Data

Untuk melakukan penambahan atau pengisian data baru, dari Input data Penyakit tersebut diklik tombol Tambah dan selanjutnya program akan menampilkan layar

Kode Penyakit	Nama Penyakit	Keterangan
P01	Asma Episodik	Asma yang mas

Gambar 39. Tampilan Tambah Data Penyakit

Setelah tombol Tambah diklik, maka secara otomatis kode gejala akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif.

Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data Penyakit.

Setelah pengisian kode Penyakit dianggap benar, selanjutnya pengguna diminta untuk menekan tombol enter pada keyboard untuk melanjutkan pengisian pada nama penyakit dan keterangan, setelah semua data telah terisi

pengguna diminta untuk menekan tombol Enter pada keyboard untuk mengaktifkan tombol Simpan seperti terlihat pada gambar

b. Koreksi Data

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data Setoran seperti pada gambar 40 di atas diklik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar berikut

Kode Penyakit	Nama Penyakit	Keterangan
P01	Asma Episodik	Asma yang mas

Gambar 40 Tambah Data Penyakit

Pada gambar 40. di atas, setelah tombol Tambah diklik, maka secara otomatis kode gejala akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif. Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan

maksud apabila pengguna ingin membatalkan pengisian data Penyakit.

Setelah pengisian kode Penyakit dianggap benar, selanjutnya pengguna diminta untuk menekan tombol enter pada keyboard untuk melanjutkan pengisian pada nama penyakit dan keterangan, setelah semua data telah terisi pengguna diminta untuk menekan tombol Enter pada keyboard untuk mengaktifkan tombol Simpan seperti terlihat pada gambar 41 berikut :

Gambar 41. Pengisian Data Penyakit

Berdasarkan gambar 41 di atas, apabila tombol Simpan diklik maka program akan menyimpan seluruh isi data yang telah diisi ke dalam database dan hasilnya terlihat pada DataGrid yang telah bertambah isinya dan tampilan kembali seperti semula seperti terlihat pada gambar 42 berikut:

Gambar 42 Tampilan Data Penyakit Yang Telah diisi

1. Koreksi Data Penyakit

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data Setoran seperti pada gambar 43 di atas diklik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar 43 berikut :

Gambar 43. Koreksi Data Penyakit

Berdasarkan gambar 43 di atas, terlihat bahwa untuk melakukan perbaikan terhadap data dilakukan dengan cara mengetik Kode Penyakit yang ingin di perbaiki.

Dalam hal ini akan dilakukan perbaikan terhadap data dengan Kode penyakit P02, kemudian program akan membandingkan kode tersebut dengan data kode Penyakit pada database, jika ditemukan maka ditampilkan isi data lengkapnya.

Apabila Kode Penyakit yang dipilih tidak ditemukan, program kemudian menampilkan pesan data tidak ditemukan, sedangkan apabila ditemukan, program menampilkannya ke layar sebagaimana terlihat pada tampilan gambar 44 sebagai berikut :



Gambar 44 Tampilan Data Penyakit yang akan dikoreksi

Dari gambar 45 di atas, diklik tombol Yes maka program selanjutnya meminta pengguna untuk mengisi data Penyakit yang lebih benar seperti tampilan gambar 45 berikut :

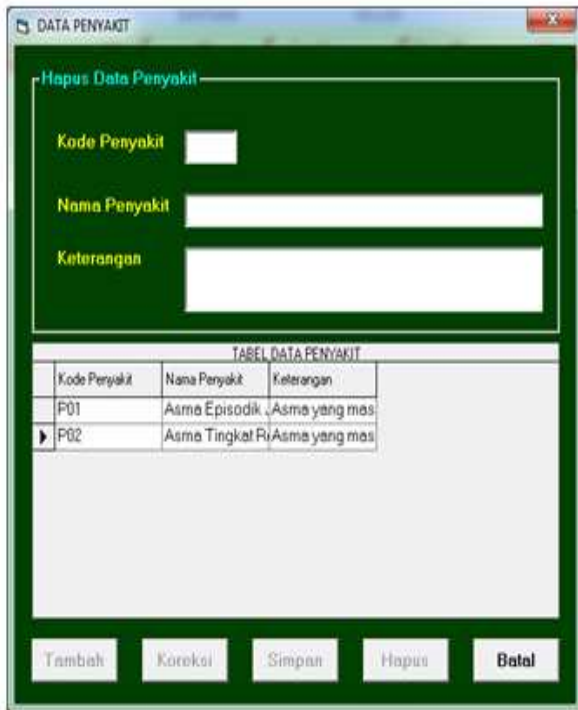


Gambar 45. Proses Perbaikan Data Penyakit

Setelah tombol Simpan pada gambar 45 di atas diklik, maka data terbaru dengan kode Penyakit disimpan dan program kemudian kembali seperti tampilan gambar 45 setelah melakukan penyimpanan terhadap data yang telah diperbaiki.

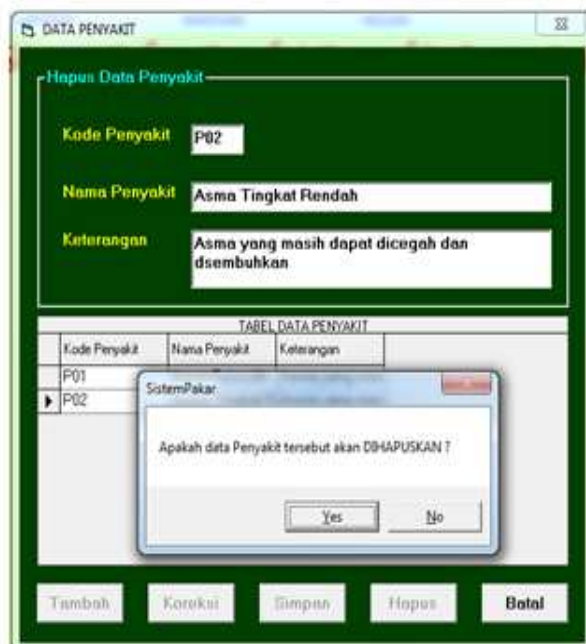
2. Hapus Data Penyakit

Berfungsi untuk dapat melakukan penghapusan terhadap data yang telah tersimpan dalam database. Apabila tombol Hapus diklik maka program selanjutnya akan memberikan tampilan seperti terlihat pada gambar 46 berikut :



Gambar 46. Tampilan Hapus Data Penyakit

Setelah Kode Penyakit diisi dan selanjutnya menekan tombol enter, maka program akan memberikan tampilan seperti gambar 47 berikut



Gambar 47 Tampilan Proses Penghapusan Data Penyakit

Berdasarkan gambar 47 di atas, apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan kode Penyakit P02 yang telah dipilih, baik pada data penyakit maupun pada databasenya.

d. Input Data Solusi

Untuk menampilkan Input Data solusi, dari Menu Utama diklik Input Data selanjutnya dengan meng-klik solusi, maka program akan memberikan tampilan program seperti gambar 48 berikut :



Gambar 48. Input Data Solusi

Berdasarkan gambar 48 di atas, dapat dijelaskan bahwa Input data Solusi berfungsi sebagai tatap muka (*interface*) guna melakukan penambahan data, perbaikan data, dan penghapusan data Solusi.

1. Tambah Data Solusi

Untuk melakukan penambahan atau pengisian data baru, dari Input data solusi tersebut diklik tombol Tambah dan selanjutnya program akan menampilkan layar seperti gambar 49 berikut

Gambar 49 Tambah Data Solusi

Pada gambar 49. di atas, tombol Tambah diklik, maka secara otomatis kode solusi akan terisi dan status beberapa tombol Tambah, Koreksi, Hapus, Simpan, dan Cetak menjadi tidak aktif. Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data Solusi. Selain itu program meminta pengguna memilih kode penyakit, nama penyakit dan penanggulangan.

Selanjutnya pengguna diminta menekan tombol Enter untuk mengaktifkan tombol Simpan sbelumnya tidak aktif akan menjadi aktif seperti terlihat pada gambar 50 berikut :

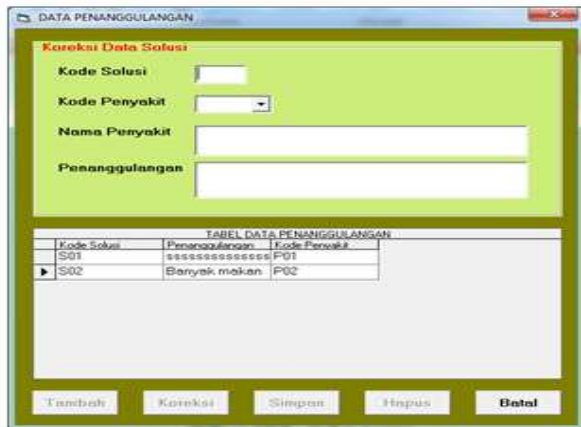
Gambar 50. Pengisian Data Solusi

Berdasarkan gambar 51 di atas, apabila tombol Simpan diklik maka program akan menyimpan seluruh isi data yang telah diisi ke dalam database dan hasilnya terlihat pada DataGridView yang telah bertambah isinya dan tombol Cetak yang tidak aktif menjadi aktif seperti terlihat pada gambar 51 berikut:

Gambar 51 Tampilan Data Solusi Yang Telah diisi

2. Koreksi Data Solusi

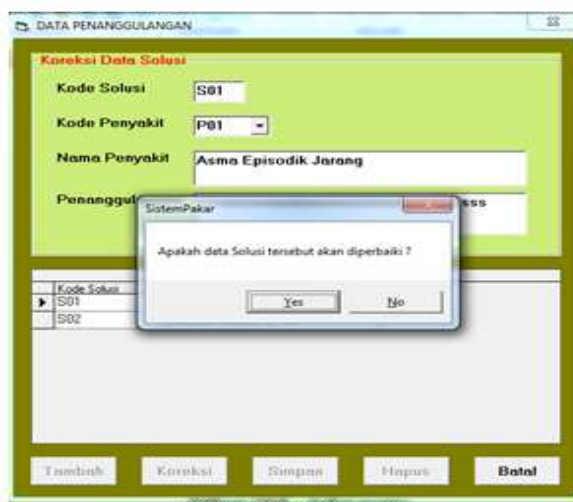
Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data solusi seperti pada gambar 4.30 di atas diklik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar 4.31 berikut :



Gambar 52 Tampilan Koreksi Data Solusi

Berdasarkan gambar 52 di atas, terlihat bahwa untuk melakukan perbaikan terhadap data dilakukan dengan cara mengetik kode solusi yang ingin di perbaiki. Dalam hal ini akan dilakukan perbaikan terhadap data dengan kode solusi S01, kemudian program akan membandingkan kode solusi tersebut dengan data kode solusi pada database, jika ditemukan maka ditampilkan isi data lengkapnya.

Apabila kode solusi yang dipilih tidak ditemukan, program kemudian menampilkan pesan data tidak ditemukan, sedangkan apabila ditemukan, program menampilkannya ke layar sebagaimana terlihat pada tampilan gambar 53 sebagai berikut:



Gambar 53 Tampilan Data solusi yang akan dikoreksi

Dari gambar 54 di atas, diklik tombol Yes maka program selanjutnya meminta pengguna untuk mengisi data Solusi yang lebih benar seperti tampilan gambar 54 berikut :



Gambar 54 Proses Perbaikan Data Solusi

Setelah tombol Simpan pada gambar 54 di atas diklik, maka data terbaru dengan kode solusi disimpan dan program kemudian kembali seperti tampilan gambar 54 setelah melakukan penyimpanan terhadap data yang telah diperbaiki.

3. Hapus Data Solusi

Berfungsi untuk dapat melakukan penghapusan terhadap data yang telah tersimpan dalam database. Apabila tombol Hapus diklik maka program selanjutnya akan memberikan tampilan seperti terlihat pada gambar 55 berikut :



Gambar 55. Tampilan Hapus Data Solusi

Setelah kode solusi diisi dan selanjutnya menekan tombol enter, maka program akan memberikan tampilan seperti gambar 56 berikut



Gambar 56 Tampilan Proses Penghapusan Data Solusi

Berdasarkan gambar 56 di atas, apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan Kode Solusi S02 yang telah dipilih, baik pada data Solusi maupun pada databasenya.

e. Input Data Rule

Untuk menampilkan Input Data Rule, dari Menu Utama diklik Input Data selanjutnya dengan mengklik rule, maka program akan memberikan tampilan program seperti gambar 57 berikut :

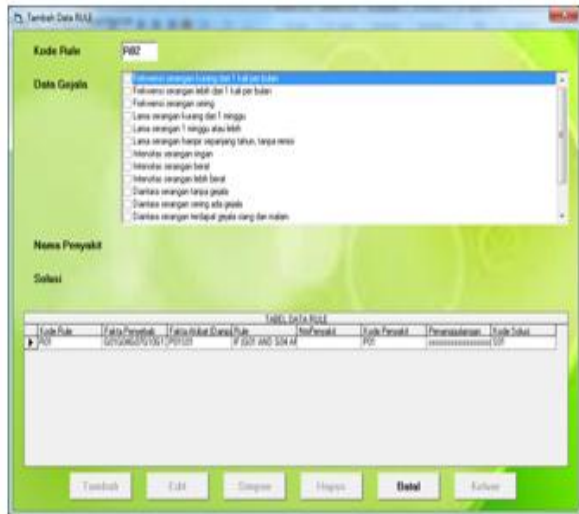


Gambar 57. Input Data Rule

Berdasarkan gambar 57 di atas, dapat dijelaskan bahwa Input data rule berfungsi sebagai tatap muka (*interface*) guna melakukan penambahan data, perbaikan data, dan penghapusan data Rule.

1. Tambah Data Rule

Untuk melakukan penambahan atau pengisian data baru, dari Input data Rule tersebut diklik tombol Tambah dan selanjutnya program akan menampilkan layar seperti gambar 58 berikut

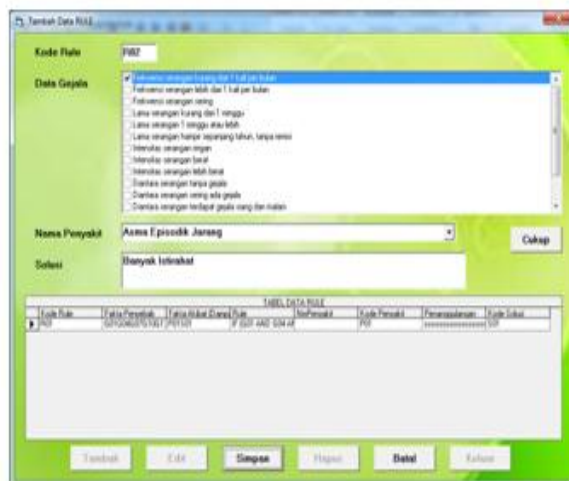


Gambar 58. Tambah Data Rule

Pada gambar 58. di atas, tombol Tambah diklik, maka secara otomatis kode rule akan terisi dan status beberapa tombol Tambah, Edit, Hapus, Simpan dan Keluar menjadi tidak aktif dan secara.

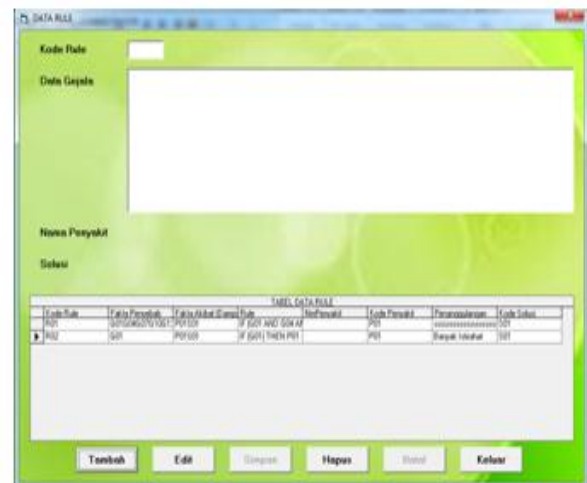
Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data Rule.

Selanjutnya program meminta pengguna untuk memilih mengisi daftar gejala, nama penyakit dan solusi. Selanjutnya dengan menekan enter pada keyboard maka tombol Simpan yang sebelumnya tidak aktif menjadi aktif seperti terlihat pada gambar 59 berikut :



Gambar 59. Pengisian Data Rule

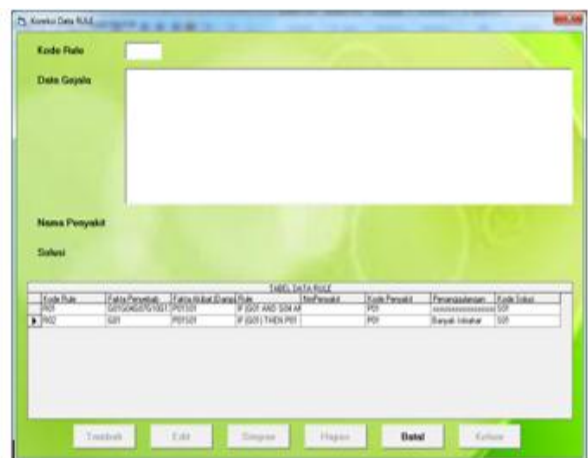
Berdasarkan gambar 60 di atas, apabila tombol Simpan diklik maka program akan menyimpan seluruh isi data yang telah diisi ke dalam database dan hasilnya terlihat pada DataGrid yang telah bertambah isinya dan tampilan kembali seperti semula seperti terlihat pada gambar 60 berikut:



Gambar 60 Tampilan Data Rule Yang Telah diisi

2. Koreksi Data Rule

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data Rule seperti pada gambar 4.39 di atas di klik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar 61 berikut :



Gambar 61 Tampilan Koreksi Data Rule

Berdasarkan gambar 4.44 di atas, apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan kode rule R02 yang telah dipilih, baik pada data rule maupun pada databasenya.

4.1.1.2 Sub Menu Laporan

Seperti telah disinggung pada awal bab ini, menu Laporan terdiri atas 3 (tiga) buah pilihan sub-menu, yakni Laporan Gejala, Daftar Penanggulangan dan Rekap Pasien Konsultasi. Berikut ini tampilan masing-masing hasil Laporan, yaitu :

a. Sub Menu Laporan Gejala

Berfungsi untuk dapat menampilkan Laporan Gejala, dari menu utama diklik pada tombol Laporan kemudian pilih sub menu Laporan Gejala, dengan tampilan seperti gambar 68 berikut :



Gambar 68. Laporan Cetak Gejala

b. Sub Menu Daftar Penanggulangan

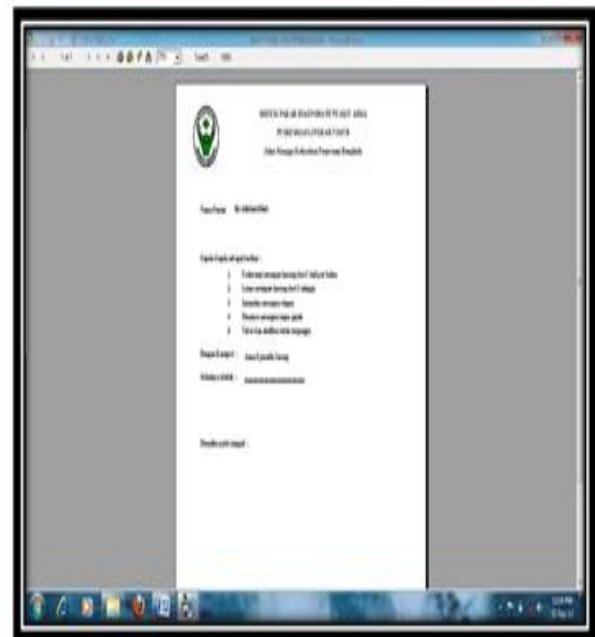
Berfungsi untuk dapat menampilkan Laporan Daftar Penanggulangan, dari menu utama diklik pada tombol Laporan kemudian pilih sub menu Daftar Penanggulangan, dengan tampilan seperti gambar 4.46 berikut :



Gambar 69 Laporan Daftar Penanggulangan

c. Sub Menu Laporan Rekap Pasien Konsultasi

Berfungsi untuk dapat menampilkan Laporan Rekap Pasien Konsultasi, dari menu utama diklik pada tombol Laporan kemudian pilih sub menu Rekap Pasien Konsultasi, dengan tampilan seperti gambar 70 berikut :



Gambar 70. Laporan Rekap Pasien Konsultasi

4.1.2 Login Pengguna

Berdasarkan gambar 71 pada awal bab, apabila pengguna maka selanjutnya program akan menampilkan menu utama seperti gambar 4.48 berikut:



Gambar 71. Tampilan Menu Utama untuk Pengguna

Berdasarkan gambar 71. di atas dapat diamati bahwa pada menu utama terdapat empat macam pilihan (Submenu), yakni Submenu Input Data, Submenu Cetak Laporan, Submenu Bantuan dan Submenu Keluar.

Submenu Input Data pada Pengguna terdiri dari beberapa pilihan lanjutan (Sub-submenu) Sub-sub menu Input Data Pasien dan Konsultasi seperti terlihat pada gambar 72. berikut :



Gambar 72. Tampilan Sub Menu Input Data untuk Pengguna

Masing-masing pilihan pada sub menu Input Data tersebut disediakan dengan tujuan untuk melakukan pengolahan data dan memberikan informasi kepada Pasien.

Submenu Cetak Hasil disediakan hanya untuk menampilkan Laporan Hasil Konsultasi Pasien.

Submenu Bantuan disediakan untuk memberikan tata cara penggunaan program, seperti terlihat pada gambar 73. berikut :



Gambar 73. Petunjuk Penggunaan Program

Sedangkan sub menu Keluar disediakan untuk melayani pengguna apabila menginginkan keluar dari program, yang mana sub menu ini bersifat tambahan saja, karena untuk menutup program dapat juga melalui fasilitas Windows yang telah ada yakni Close atau dengan shortcut-key (Alt+F4).

Berikut ini akan diuraikan lebih lanjut dan rinci dari masing-masing submenu sesuai dengan kepentingannya dengan maksud untuk mempertegas proses pemakaian program yang sekaligus merupakan panduan penggunaan.

4.1.1 Sub Menu Input Data

Sub menu Input Data disediakan untuk melayani pengelolaan terhadap data yang akan dikerjakan. Dimana sub menu data ini memiliki 2 (dua) macam pilihan, yakni Sub-sub Input Data

Pasien dan Konsultasi, berikut ini akan dibahas masing-masing pilihan, yaitu :

a. Input Data Pasien

Untuk menampilkan Input Data Pasien, dari Menu Utama diklik Input Data selanjutnya

dengan meng-klik Pasien, maka program akan memberikan tampilan program seperti gambar 74 berikut :

DATA PASIEN

Kode Pasien

Nama Pasien

Alamat

Umur Tahun

Jenis Kelamin

TABEL DATA PASIEN				
Id.Pasien	Nama.Pasien	Alamat	Umur	Jenis.Kel
00001	lkj dslkjds\lkds:sdh kjhdskjhdsk	23		Laki-Laki

Tambah Koreksi Simpan Hapus Keluar

Gambar 74. Input Data Pasien

Berdasarkan gambar 74 di atas, dapat dijelaskan bahwa Input data Pasien berfungsi sebagai tatap muka (*interface*) guna melakukan penambahan data, perbaikan data, dan penghapusan data Pasien.

1. Tambah Data Pasien

Untuk melakukan penambahan atau pengisian data baru, dari Input data Pasien tersebut diklik tombol Tambah dan selanjutnya program akan menampilkan layar seperti gambar 75 berikut :

DATA PASIEN

Kode Pasien: 00002

Nama Pasien:

Alamat:

Umur: Tahun

Jenis Kelamin:

Kode Pasien	Nama Pasien	Alamat	Umur	Jenis Kelamin
00001	Iq dslkj dsjkl dsj k jhdskjhdskj23			Laki-Laki

Tambah Kembali Simpan Hapus Batal

Gambar 75. Tambah Data Pasien

Pada gambar 75. di atas, setelah tombol Tambah diklik, maka secara otomatis kode pasien akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif. Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data pasien.

Setelah pengisian kode pasien dianggap benar, selanjutnya program meminta pengguna menekan tombol Enter pada keyboard untuk melanjutkan pada pengisian nama pasien, alamat, umur, dan jenis kelamin. Setelah pengisian seluruh data selesai pengguna diminta menekan tombol Enter untuk mengaktifkan tombol Simpan seperti terlihat pada gambar 76 berikut :

Gambar 77. Pengisian Data Pasien

Berdasarkan gambar 77 di atas, apabila tombol Simpan diklik maka program akan menyimpan seluruh isi data yang telah diisi ke dalam database dan hasilnya terlihat pada Tabel Data Pasien yang telah bertambah isinya pada Datagrid dan tampilan kembali seperti semula seperti terlihat pada gambar 78 berikut:

Gambar 78. Tampilan Data Pasien Yang Telah diisi

2. Koreksi Data Pasien

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data pasien seperti pada gambar 79 di atas diklik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar 79 berikut :

Gambar 79. Tampilan Koreksi Data Pasien

Berdasarkan gambar 79 di atas, terlihat bahwa untuk melakukan perbaikan terhadap data dilakukan dengan cara mengetik kode pasien yang ingin di perbaiki.

Dalam hal ini akan dilakukan perbaikan terhadap data dengan kode pasien 00001, kemudian program akan membandingkan kode tersebut dengan data kode pasien pada database, jika ditemukan maka ditampilkan isi data lengkapnya.

Apabila kode pasien yang dipilih tidak ditemukan, program kemudian menampilkan pesan data tidak ditemukan, sedangkan apabila ditemukan, program menampilkannya ke layar sebagaimana terlihat pada tampilan gambar 80 sebagai berikut:



Gambar 80 Tampilan Data pasien yang akan dikoreksi

Dari gambar 80 di atas, diklik tombol Yes maka program selanjutnya meminta pengguna untuk mengisikan data pasien yang lebih benar seperti tampilan gambar 81 berikut :



Gambar 81. Proses Perbaikan Data Pasien

Setelah tombol Simpan pada gambar 81 di atas diklik, maka data terbaru dengan kode pasien 00001 telah disimpan dan program kemudian kembali seperti tampilan gambar 81 setelah melakukan penyimpanan terhadap data yang telah diperbaiki.

3. Hapus Data Pasien

Berfungsi untuk dapat melakukan penghapusan terhadap data yang telah tersimpan dalam database.

Apabila tombol Hapus diklik maka program selanjutnya akan memberikan tampilan seperti terlihat pada gambar 82 berikut :



Gambar 82 Tampilan Hapus Data Pasien

Setelah kode Pasien diisi dan selanjutnya menekan tombol enter, maka program akan memberikan tampilan seperti gambar 83 berikut



Gambar 83 Tampilan Proses Penghapusan Data Pasien

Berdasarkan gambar 83 di atas, apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan kode gejala 00001 yang telah dipilih, baik pada data Pasien maupun pada databasenya.

c. Input Data Konsultasi

Untuk menampilkan Input Data Konsultasi, dari Menu Utama diklik Input Data selanjutnya dengan meng-klik Konsultasi, maka program akan memberikan tampilan program seperti gambar 84 berikut



Gambar 84. Input Data Konsultasi

Berdasarkan gambar 84 di atas, dapat dijelaskan bahwa Input data konsultasi berfungsi sebagai tatap muka (*interface*) guna melakukan penambahan data, perbaikan data, dan penghapusan data konsultasi.

1. Tambah Data Konsultasi

Untuk melakukan penambahan atau pengisian data baru, dari Input data Konsultasi tersebut diklik tombol Tambah dan selanjutnya program akan menampilkan layar seperti gambar 85 berikut :



Gambar 85. Tambah Data Konsultasi

Pada gambar 85. di atas, setelah tombol Tambah diklik, maka secara otomatis kode konsultasi akan langsung terisi dan status beberapa tombol Tambah, Koreksi, Simpan dan Hapus menjadi tidak aktif.

Sedangkan tombol Batal yang sebelumnya tidak aktif menjadi aktif dengan maksud apabila pengguna ingin membatalkan pengisian data pasien.

Setelah pengisian kode konsultasi dianggap benar, selanjutnya program meminta pengguna menekan tombol Enter pada keyboard untuk melanjutkan pada pengisian kode pasien, nama pasien, jenis kelamin dan umur.

Setelah pengisian seluruh data selesai pengguna diminta menekan tombol Enter untuk mengisi daftar gejala, selanjutnya menekan tombol cukup untuk mengakhiri pemilihan, maka nama penyakit yang diderika dan solusinya telah berhasil di dapat seperti gambar 86 berikut:

[illegible]

Gambar 86. Pengisian Data Konsultasi

Berdasarkan gambar 86 di atas, apabila tombol Simpan diklik maka program akan menyimpan seluruh isi data yang telah diisi ke dalam database dan hasilnya terlihat pada Tabel Data konsultasi yang telah bertambah isinya dalam Datagrid dan tampilan kembali seperti semula seperti terlihat pada gambar 86 .

2. Koreksi Data konsultasi

Berfungsi Untuk dapat melakukan perbaikan terhadap data yang telah tersimpan dalam database, maka dari Data konsultasi seperti pada gambar 87 di atas diklik tombol Koreksi maka program akan menampilkan seperti terlihat pada gambar 87 berikut :

Koreksi Data Konsultasi

Kode Konsultasi

Tanggal Konsultasi 22-Sep-12

Kode Pasien

Nama Pasien

Jenis Kelamin

Usia

Tahun

Detail Gejala

Indikasi Penyakit

Staf

Tambah Edit Hapus Simpan Batal

Gambar 87. Tampilan Koreksi Data konsultasi

Berdasarkan gambar 87 di atas, terlihat bahwa untuk melakukan perbaikan terhadap data dilakukan dengan cara mengetik kode konsultasi yang ingin di perbaiki.

Dalam hal ini akan dilakukan perbaikan terhadap data dengan kode konsultasi 000001, kemudian program akan membandingkan kode tersebut dengan data kode konsultasi pada database, jika ditemukan maka ditampilkan isi data lengkapnya.

Apabila kode konsultasi yang dipilih tidak ditemukan, program kemudian menampilkan pesan data tidak ditemukan, sedangkan apabila ditemukan, program menampilkannya ke layar sebagaimana terlihat pada tampilan gambar 88 sebagai berikut:

Kode Konsultasi: 000001 Tanggal Konsultasi: 27-Sep-12

Kode Pasien: 00001 Nama Pasien: isi

Jenis Kelamin: Perempuan Usia: 73 Tahun: isi

Daftar Gejala

- ☒ Lama serangan kurang dari 1 minggu
- ☐ Lama serangan 1 minggu atau lebih
- ☐ Lama serangan hampir sepanjang tahun, hampir terus-menerus
- ☒ Intensitas serangan ringan
- ☐ Intensitas serangan berat
- ☐ Intensitas serangan tidak teratur
- ☒ Disertai sakit kepala
- ☐ Disertai sakit dada
- ☐ Disertai sakit perut
- ☒ Disertai sakit lainnya

Indikasi Penyakit Asma & penyakit paru

Solusi inhalasi kortikosteroid

DAFTAR DATA KONSULTASI

No	Tanggal	Nama Pasien	Tanggal Konsultasi	Gejala	Pemeriksaan Fisik	Diagnosis	Tgl. Rujukan
000001	27-Sep-12	isi	27-Sep-12	Asma & penyakit paru	Gejala: sesak, batuk, wheezing	Asma & penyakit paru	27-Sep-12
000002	27-Sep-12	isi	27-Sep-12	Asma & penyakit paru	Gejala: sesak, batuk, wheezing	Asma & penyakit paru	27-Sep-12
000003	27-Sep-12	isi	27-Sep-12	Asma & penyakit paru	Gejala: sesak, batuk, wheezing	Asma & penyakit paru	27-Sep-12

Tombol: Kembali, Edit, Simpan, Hapus, Batal

Gambar 88 Tampilan Data konsultasi yang akan dikoreksi

Dari gambar 88 di atas, diklik tombol Yes maka program selanjutnya meminta pengguna untuk mengisi data konsultasi yang lebih benar seperti tampilan gambar 89 berikut :

Kode Konsultasi: 000001 Tanggal Konsultasi: 22-Sep-12

Kode Pasien: 000001 Nama Pasien: Jns

Jenis Kelamin: Perempuan Umur: 23 Tahun

Daftar Gejala

- ☒ Lama serangan kurang dari 1 minggu
- ☐ Lama serangan 1 minggu atau lebih
- ☐ Lama serangan hampir sepanjang tahun, tanpa remisi
- ☒ Intensitas serangan ringan
- ☐ Intensitas serangan berat
- ☐ Intensitas serangan lebih berat
- ☒ Di antara serangan tanpa gejala
- ☐ Di antara serangan sering ada gejala
- ☐ Di antara serangan terdapat gejala sedang dan malam
- ☒ Tidak ada riwayat asma sebelumnya

Indikasi Penyakit Asma Epistatik Jantung

Gejala banyak banyak istirahat

Kode Konsultasi	Nama Pasien	Jenis Kelamin	Umur	Tanggal Konsultasi	Status
000001	Jns	Perempuan	23	22-Sep-12	00000000000000000000

Tombol: Tambah, Edit, Simpan, Hapus, Batalkan

Gambar 89. Proses Perbaikan Data konsultasi

Setelah tombol Simpan pada gambar 89 di atas diklik, maka data terbaru dengan kode pasien 000001 telah disimpan dan program kemudian kembali seperti tampilan gambar 89 setelah melakukan penyimpanan terhadap data yang telah diperbaiki.

3. Hapus Data konsultasi

Berfungsi untuk dapat melakukan penghapusan terhadap data yang telah tersimpan dalam database. Apabila tombol Hapus diklik maka program selanjutnya akan memberikan tampilan seperti terlihat pada gambar 90 berikut :

Kode Konsultasi: Tanggal Konsultasi: 22-Sep-12

Kode Pasien: Nama Pasien: Jns

Jenis Kelamin: Perempuan Umur: 23 Tahun

Daftar Gejala

- ☒ Lama serangan kurang dari 1 minggu
- ☐ Lama serangan 1 minggu atau lebih
- ☐ Lama serangan hampir sepanjang tahun, tanpa remisi
- ☒ Intensitas serangan ringan
- ☐ Intensitas serangan berat
- ☐ Intensitas serangan lebih berat
- ☒ Di antara serangan tanpa gejala
- ☐ Di antara serangan sering ada gejala
- ☐ Di antara serangan terdapat gejala sedang dan malam
- ☒ Tidak ada riwayat asma sebelumnya

Indikasi Penyakit Asma Epistatik Jantung

Gejala banyak banyak istirahat

Kode Konsultasi	Nama Pasien	Jenis Kelamin	Umur	Tanggal Konsultasi	Status
000001	Jns	Perempuan	23	22-Sep-12	00000000000000000000

Tombol: Tambah, Edit, Simpan, Hapus, Batalkan

Gambar 90 Tampilan Hapus Data konsultasi

Setelah kode konsultasi diisi dan selanjutnya menekan tombol enter, maka program akan memberikan tampilan seperti gambar 91 berikut

Kode Konsultasi: 000002 Tanggal Konsultasi: 22-Sep-12

Kode Pasien: 000001 Nama Pasien: Jns

Jenis Kelamin: Perempuan Umur: 23 Tahun

Daftar Gejala

- ☒ Lama serangan kurang dari 1 minggu
- ☐ Lama serangan 1 minggu atau lebih
- ☐ Lama serangan hampir sepanjang tahun, tanpa remisi
- ☒ Intensitas serangan ringan
- ☐ Intensitas serangan berat
- ☐ Intensitas serangan lebih berat
- ☒ Di antara serangan tanpa gejala
- ☐ Di antara serangan sering ada gejala
- ☐ Di antara serangan terdapat gejala sedang dan malam
- ☒ Tidak ada riwayat asma sebelumnya

Indikasi Penyakit Asma Epistatik Jantung

Gejala banyak banyak istirahat

Apakah Data Konsultasi tersebut akan DIHAPUSKAN?

Tombol: Tambah, Edit, Simpan, Hapus, Batalkan

Gambar 91 Tampilan Proses Penghapusan Data konsultasi

Berdasarkan gambar 91 di atas, apabila tombol Yes diklik maka program selanjutnya akan menghapus data dengan kode gejala 000002 yang telah dipilih, baik pada data konsultasi maupun pada databasenya.

4.1.2.1 Cetak Hasil

Berfungsi untuk dapat menampilkan Laporan Konsultasi, dari menu utama diklik pada tombol cetak hasil maka program akan memberikan tampilan seperti gambar 92 berikut



Gambar 92 Cetak hasil konsultasi

Berdasarkan gambar diatas, apabila tombol mulai di klik maka program akan menampilkan tampilan seperti gambar 93 berikut



Gambar 93 laporan hasil konsultasi

4.1. Pengujian Sistem

Proses pengujian sistem yang telah dibangun selanjutnya diimplementasi pada tempat penelitian dan diuji coba. Dalam proses implementasi yang dilakukan, program dipasang pada sistem operasi MS Windows dengan spesifikasi perangkat keras yang sesuai dengan proposal tedahulu.

Pengujian sistem baru ini dilaksanakan pada Kantor Puskesmas Lingkar Timur. Dimana implementasian ini dilakukan Karyawan-karyawti puskesmas, setelah penulis menerangkan dan menjelaskan tata cara menjalankan program ini.

Selama perancangan program penyusun banyak meminta masukan dari pihak puskesmas Lingkar Timur untuk mencatat berbagai ketentuan yang diberlakukan oleh pihak puskesmas Lingkar Timur. Dengan demikian program hasil yang telah dibangunpun diharapkan memberikan informasi tentang kesehatan pasien.

4.2. Perbandingan Sistem Lama dan Sistem Baru

Adapun jenis keuntungan penggunaan sistem baru dibandingkan sistem lama yang dimaksud dapat dijelaskan sebagai berikut :

Table 4.1 Perbandingan Sistem Lama dan Sistem Baru

Jenis	Sistem Lama	Sistem Baru
Ketelitian	Ketelitian sangat rendah karena menyangkut daya ingat bagian pembukuan dan tidak otomatis	Ketelitian tinggi karena sistem telah terintegrasi antar tabel yang terkoneksi
Keamanan Data	Sangat terbuka kemungkinan terjadi kehilangan data	Sangat aman karena data hasil proses tercakup dalam database.
Kebutuhan Media Simpan	Sangat besar dalam bentuk map-map dan perlengkapan administrasi kantor lainnya	Simpel dan hanya memerlukan ruang simpan yang kecil (Flash Disk, Hard Disk)

V. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pengamatan yang telah dilakukan melalui implementasi dan penerapan pemakaian program pada puskesmas Lingkar Timur, dapat disimpulkan sebagai berikut :

1. Keberadaan program sebagai alat bantu memberikan informasi tentang Cetak hasil konsultasi pasien.
2. Aplikasi Aplikasi Sistem Pakar Deteksi Penyakit Asma Pada Puskesmas Lingkar Timur ini mampu kemudahan dalam penyelesaian pekerjaan.
3. Dengan adanya program ini, manajemen dan kinerja puskesmas menjadi lebih efektif dan efisien.

4. Dengan adanya program ini pada Puskesmas Lingkar Timur, mampu memberikan kontribusi positif untuk peningkatan kinerja manajemen.

5.2. Saran

Berdasarkan pembahasan dan kesimpulan yang telah diuraikan di atas, saran dan rekomendasi yang dapat diberikan adalah :

1. Agar Aplikasi Sistem Pakar Deteksi Penyakit Asma Pada Puskesmas Lingkar Timur lebih dikenal secara luas, maka perlu kiranya adanya program yang mampu diakses secara On-line dari manapun yang dalam hal ini dapat diwakili oleh Website Puskesmas Linkar Timur
2. Untuk kemudian hari diharapkan dapat dilakukan pengembangan aplikasi yang dapat dilakukan menggunakan aplikasi lain seperti java yang dapat diakses melalui handphone ataupun aplikasi lainnya.

Sinarmata & Paryudi, 2007, "*Perancangan Basis Data*", Penerbit Andi Yogyakarta, 276 Halaman.

T. Sutojo, Edy Mulyanto, Vincent Suhartono, 2011. Kecerdasan Buatan. Penerbit Andi. Yogyakarta.

Andri Kristanto, 2008. Perancangan Sistem Informasi dan Aplikasinya. Gava Media, 171 halaman.

Arahmi, Muhammad. 2004. Konsep Dasar Sistem Pakar: Andi, Yogyakarta, 199 halaman.

Wikipedia, 2010. Sistem Informasi. id.wikipedia.org/wiki/Sistem_informasi

DAFTAR PUSTAKA

- Anita D. dan Muhammad A., 2004. Konsep Kecerdasan Buatan. Penerbit Andi. Yogyakarta.
- Anonim, 2005. Asma. PT. Gramedia Pustaka Utama. Jakarta, 180 halaman.
- Barza Ahmad, 2006. Diktat Mata Kuliah kecerdasan Buatan. Universitas Gajah Mada. Yogyakarta.
- Negnevitsky, M., 2002. Artificial Intelligence – A Guide to Intelligent Systems. First Edition. Addison – Wesley. ©Peason education Limited 2002.. United Kingdom..
- Rolston, D.W., 1988. Principles of Artificial Intelligence and Expert Systems Development. McGraw-Hill International Editions. McGraw-Hill Book Company. Singapore.